

# Desenvolupament d'una aplicació Android per a ciutats mitjanes



Portal Badalona

**Memòria final**  
**Jordi Rubiralta Cortés**

Quadrimestre 1 2019-2020  
Directora: Silvia Llorente Viejo  
Grau en Enginyeria Informàtica  
Especialització en Tecnologies de la Informació



UNIVERSITAT POLITÈCNICA DE CATALUNYA  
BARCELONATECH

Facultat d'Informàtica de Barcelona



## Agraïments

En primer lloc, m'agradaria agrair a la professora Silvia Llorente per haver acceptat dirigir aquest projecte i haver-me aconsellat i ajudat en el desenvolupament.

En segon lloc, als polítics que van dedicar uns minuts a parlar amb mi, que es van interessar pel meu treball i que em van donar la seva opinió sobre aquest tema.

Finalment, agrair a la meva cosina Laia per haver-me ajudat amb el disseny del logotip, en la selecció de colors i en la tipografia de l'aplicació.

## Resum

Cada dia les tecnologies avancen i estan més presents al nostre dia a dia. És per això que les ciutats intel·ligents cada cop estan agafant més importància al món de la tecnologia i cada cop més ciutats es disposen a invertir-hi. Donada aquesta situació, s'ha plantejat aquest projecte com una introducció a les ciutats intel·ligents per a ciutats mitjanes.

Les funcionalitats d'aquesta aplicació són mostrar notícies i esdeveniments de la ciutat, reportar incidències per tal que l'ajuntament en tingui constància i respondre enquestes proposades per l'ajuntament. D'aquesta manera, es facilita la comunicació entre els ciutadans i l'ajuntament amb la finalitat d'aconseguir entre tots una ciutat millor.

## Resumen

Cada día las tecnologías avanzan y están más presentes en nuestro día a día. Es por eso que las ciudades inteligentes cada vez cogen más importancia en el mundo de la tecnología y cada vez más ciudades se disponen a invertir en ello. Dada esta situación, se ha planteado este proyecto como una introducción a las ciudades inteligentes para ciudades medianas.

Las funcionalidades de esta aplicación son mostrar noticias y eventos de la ciudad, reportar incidencias para que el ayuntamiento tenga constancia y resolver encuestas propuestas por el ayuntamiento. De esta manera, se facilita la comunicación entre los ciudadanos y el ayuntamiento con la finalidad de conseguir entre todos una ciudad mejor.

## Abstract

Technologies are advancing every day and they are present in our daily lives more than ever before. This is why smart cities are becoming more and more important in the world of technology and more cities are investing in it. Given this situation, this project has been proposed as an introduction to smart cities for medium-sized cities.

The functionalities of this application are to show news and events going on in the city, to report incidents for the city council to record, and to answer surveys proposed by the city council. This should facilitate communication between the citizens and the city council in order to achieve a better city.

# Índex

<b>1.- INTRODUCCIÓ .....</b>	<b>10 -</b>
1.1.- DESCRIPCIÓ DEL PROJECTE .....	10 -
1.2.- FORMULACIÓ DEL PROBLEMA .....	10 -
1.3.- OBJECTIUS DEL PROJECTE .....	11 -
1.4.- CONTEXT .....	12 -
1.5.- STAKEHOLDERS .....	13 -
1.6.- ABAST .....	14 -
1.6.1.- Abast del projecte .....	14 -
1.6.2.- Possibles obstacles .....	15 -
1.7.- ROL PERSONAL DEL PROJECTE .....	16 -
<b>2.- ESTAT DE L'ART .....</b>	<b>17 -</b>
2.1.- TECNOLOGIES EXISTENTS .....	17 -
2.1.1.- Gestió de tasques .....	17 -
2.1.2.- Gestió de codi .....	17 -
2.1.3.- Arquitectura de l'aplicació mòbil .....	18 -
2.1.4.- Desenvolupament front-end .....	19 -
2.1.5.- Desenvolupament back-end .....	20 -
2.1.6.- Desenvolupament web .....	20 -
2.1.7.- Administració de bases de dades .....	21 -
2.1.8.- Documentació .....	22 -
2.2.- TECNOLOGIES SELECCIONADES .....	22 -
2.3.- INTEGRACIÓ DE CONEIXEMENTS .....	23 -
<b>3.- DESENVOLUPAMENT DEL PROJECTE .....</b>	<b>25 -</b>
3.1.- TREBALL PREVI AL DESENVOLUPAMENT .....	25 -
3.2.- BASE DE DADES .....	26 -
3.3.- SERVIDOR BACK-END .....	27 -
3.3.1.- Arquitectura .....	27 -
3.3.2.- Llibreries utilitzades .....	28 -
3.3.3.- Serveis .....	29 -
3.4.- APLICACIÓ ANDROID .....	36 -
3.4.1.- Arquitectura .....	36 -
3.4.2.- Llibreries utilitzades .....	37 -
3.4.3.- Pantalles .....	38 -
3.5.- DESENVOLUPAMENT WEB .....	52 -
3.5.1.- Arquitectura .....	53 -
3.5.2.- Llibreries utilitzades .....	53 -
3.5.3.- Pantalles .....	54 -
3.6.- REPOSITORIS DEL PROJECTE .....	62 -
<b>4.- METODOLOGIA .....</b>	<b>63 -</b>
4.1.- ORGANITZACIÓ DE TASQUES .....	63 -
4.2.- ORGANITZACIÓ DE CODI .....	64 -
4.3.- GESTIÓ DE BASES DE DADES .....	65 -
<b>5.- PLANIFICACIÓ TEMPORAL .....</b>	<b>66 -</b>
5.1.- FASES DEL PROJECTE .....	66 -

5.1.1.- Desenvolupament Android .....	- 66 -
5.1.2.- Desenvolupament de la base de dades .....	- 67 -
5.1.3.- Desenvolupament de la pàgina web .....	- 68 -
5.1.4.- Desenvolupament dels serveis REST.....	- 68 -
5.1.5.- Gestió del projecte.....	- 69 -
5.2.- SEQÜÈNCIA LÒGICA .....	- 69 -
5.3.- TAULA RESUM .....	- 70 -
5.4.- ESTIMACIONS TEMPORALS I GANTT.....	- 72 -
5.5.- DESVIACIONS I PLANS ALTERNATIU .....	- 73 -
5.6.- DESVIACIONS PATIDES DURANT EL PROJECTE .....	- 74 -
<b>6.- GESTIÓ ECONÒMICA.....</b>	<b>- 75 -</b>
6.1.- PRESSUPOST .....	- 75 -
6.1.1.- Identificació dels costos.....	- 75 -
6.1.2.- Estimació dels costos.....	- 77 -
6.1.3.- Cost total .....	- 82 -
<b>7.- IDENTIFICACIÓ DE LLEIS I REGULACIONS .....</b>	<b>- 84 -</b>
7.1.- LLEI ORGÀNICA DE PROTECCIÓ DE DADES I GARANTIA DE DRETS DIGITALS. ....	- 84 -
7.2.- LLEI DE PROPIETAT INTEL·LECTUAL .....	- 85 -
<b>8. INFORME DE SOSTENIBILITAT .....</b>	<b>- 86 -</b>
8.1. AUTOAVALUACIÓ .....	- 86 -
8.2. DIMENSIÓ ECONÒMICA .....	- 86 -
8.3. DIMENSIÓ AMBIENTAL .....	- 87 -
8.4. DIMENSIÓ SOCIAL.....	- 87 -
<b>9.- CONCLUSIONS .....</b>	<b>- 89 -</b>
9.1.- DIFICULTATS I PROBLEMES .....	- 89 -
9.2. CONCLUSIONS DELS OBJECTIUS INICIALS .....	- 90 -
9.3.- CONCLUSIONS PERSONALS .....	- 90 -
9.4.- CONCLUSIONS DE COMPETÈNCIES TÈCNIQUES .....	- 91 -
9.4.1.- CTI1.1.....	- 91 -
9.4.2.- CTI.1.3.....	- 91 -
9.4.3.- CTI3.1.....	- 91 -
9.4.4.- CTI4.....	- 92 -
<b>10.- VOCABULARI .....</b>	<b>- 93 -</b>
<b>11.- REFERÈNCIES .....</b>	<b>- 95 -</b>

## Índex de figures

Figura 1. Diagrama UML de base de dades. ....	- 27 -
Figura 2. Captura de pantalla de l'Splash Screen .....	- 38 -
Figura 3. Captura de pantalla d'iniciar sessió .....	- 39 -
Figura 4. Captura de pantalla del registre erroni .....	- 40 -
Figura 5. Captura de pantalla del registre .....	- 41 -
Figura 6. Captura de pantalla del registre amb errors .....	- 42 -
Figura 7. Captura de pantalla del menú .....	- 43 -
Figura 8. Captura de pantalla de notícies.....	- 44 -
Figura 9. Captura de pantalla d'ajuda de notícies .....	- 45 -
Figura 10. Captura de pantalla d'esdeveniments.....	- 46 -
Figura 11. Captura de pantalla de les enquestes .....	- 47 -
Figura 12. Captura de pantalla de les incidències .....	- 48 -
Figura 13. Captura de pantalla de nova incidència .....	- 49 -
Figura 14. Captura de pantalla de nova incidència amb errors.....	- 50 -
Figura 15. Captura de pantalla de configuració .....	- 51 -
Figura 16. Captura de pantalla de tancar sessió.....	- 52 -
Figura 17. Captura de pantalla d'iniciar sessió (web).....	- 54 -
Figura 18. Captura de pantalla del missatge d'error d'usuari inexistent (web) .....	- 54 -
Figura 19. Captura de pantalla el missatge d'error de contrasenya incorrecta .....	- 55 -
Figura 20. Captura de pantalla de llista d'esdeveniments (web) .....	- 55 -
Figura 21. Captura de pantalla de la barra de navegació d'organitzador no administrador (web) .....	- 55 -
Figura 22. Captura de pantalla del missatge d'esdeveniment actualitzat (web).....	- 56 -
Figura 23. Captura de pantalla del missatge d'esdeveniment esborrat (web).....	- 56 -

Figura 24. Captura de pantalla de la llista d'esdeveniments quan és buida (web) ...	- 56 -
Figura 25. Captura de pantalla d'editar esdeveniment (web).....	- 57 -
Figura 26. Captura de pantalla d'afegir un esdeveniment .....	- 57 -
Figura 27. Captura de pantalla de llista d'incidències .....	- 58 -
Figura 28. Captura de pantalla de llista d'organitzadors (web).....	- 59 -
Figura 29. Captura de pantalla del missatge d'organitzador registrat (web) .....	- 59 -
Figura 30. Captura de pantalla del missatge d'organitzador esborrat (web) .....	- 59 -
Figura 31. Captura de pantalla d'afegir un organitzador (web) .....	- 60 -
Figura 32. Captura de pantalla de llista d'enquestes (web) .....	- 60 -
Figura 33. Captura de pantalla del missatge d'enquesta esborrada (web) .....	- 61 -
Figura 34. Captura de pantalla de llista d'enquestes buida (web) .....	- 61 -
Figura 35. Captura de pantalla del missatge d'enquesta creada (web) .....	- 61 -
Figura 36. Diagrama de Gantt .....	- 72 -



## Índex de taules

Taula 1. Taula de tasques definides per hores, dependències i recursos. ....	- 71 -
Taula 2. Taula de costos en relació amb les tasques.....	- 81 -
Taula 3. Taula de costos indirectes. ....	- 82 -
Taula 4. Taula de costos totals. ....	- 83 -

## 1.- Introducció

Tal i com informen les principals empreses tecnològiques i diferents estudis, el creixement dels dispositius *smartphone* està en creixement des de la seva aparició. Segons indicava *El País* (Ramón Muñoz, 2019) [1] en un article relacionat amb el *Mobile World Congress (MWC)*, el 2025 s'espera un creixement en usuaris d'*smartphones* fins al 60% de la població, és a dir, 5.000 milions d'usuaris. Per tant, és evident que una gran part de la població té al seu abast un dispositiu *smartphone*.

### 1.1.- Descripció del projecte

Aquest projecte presenta una aplicació mòbil, únicament desenvolupada pel sistema operatiu Android, que té la finalitat d'apropar els ciutadans de ciutats mitjanes amb els respectius ajuntaments per tal de facilitar-ne la comunicació.

Per tant, l'objectiu d'aquesta aplicació és oferir un software on els ciutadans podran:

- Reportar incidències del mobiliari urbà per tal que es resolguin el més aviat possible.
- Respondre enquestes plantejades per l'ajuntament per tal de donar visibilitat a l'opinió pública sobre els temes tractats als plens.
- Veure un calendari amb els esdeveniments pròxims i destacats que es duguin a terme a la ciutat
- Llegir les notícies locals més destacades.

### 1.2.- Formulació del problema

El projecte pretén enllaçar el creixement de les noves tecnologies i les ciutats mitjanes. Per entendre aquesta segmentació en el terme ciutat, és important diferenciar les ciutats mitjanes de les ciutats rurals i les grans ciutats.

És evident que les ciutats mantenen una diferència social, econòmica i urbanística segons la seva dimensió i nombre d'habitants. Les ciutats mitjanes mostren unes

característiques concretes: una identitat pròpia [2], economia circular, diversitat urbana i mobilitat sostenible [3].

Un dels principals problemes dels ajuntaments és el seu interès polític propi quan, el principal objectiu, hauria de ser vetllar per la millora de la ciutat i el benestar dels habitants. És per això que en el nostre dia a dia podem veure una sèrie de problemàtiques que són comunes a la majoria de municipis mitjans i que són fruit d'una manca evident d'abast i comunicació, com ara:

- Incidències que triguen dies o inclús mesos a resoldre's i que són de caràcter prioritari. Per exemple, un semàfor que no funciona.
- Decisions que pren l'ajuntament que no agraden a la majoria dels habitants.
- Esdeveniments amb poca ressonància que podrien interessar a un públic molt més extens.
- Desconeixement de notícies interessants sobre la ciutat.

Tot i que aquestes problemàtiques poden ser comunes també amb les grans ciutats, ens centrem únicament en ciutats mitjanes perquè ciutats com Barcelona o Madrid són suficientment grans i amb molta diversitat, de forma que poden tenir diverses aplicacions per diferents públics, ja sigui diferenciant barris, ciutadans o turistes, finalitat de l'aplicació (gestió o informativa), entre d'altres.

Per altra banda, les ciutats petites o rurals no tenen suficients habitants com per amortitzar el preu de desenvolupar i mantenir una aplicació i, possiblement, no tindran un impacte major que la comunicació de boca en boca habitual en municipis petits.

### 1.3.- Objectius del projecte

L'objectiu d'aquest projecte consisteix en unificar les diverses aplicacions que pot tenir una gran ciutat en una sola aplicació. El desenvolupament d'aquesta aplicació es pot dividir en quatre parts.

En primer lloc, s'ha de desenvolupar el *front-end* de l'aplicació, és a dir, la interfície d'usuari. Aquesta part consisteix en el desenvolupament de l'aplicació *Android*.

En segon lloc, és necessari crear i gestionar la base de dades amb tota la informació necessària dels usuaris, incidències, esdeveniments, enquestes i notícies per tal que l'aplicació pugui emmagatzemar totes aquestes dades.

En tercer lloc, és necessari desenvolupar un *back-end*, és a dir, un servidor *REST*, per tal de tenir uns serveis que permetin a l'aplicació *Android* obtenir les dades necessàries de la base de dades.

Finalment, tot i que aquest punt no és del tot imprescindible, també es marca com a objectiu ja que seria interessant que hi hagués una pàgina web simple per a usuaris administradors per tal que, l'ajuntament en qüestió, pugui donar accés a diferents organitzacions de la ciutat per poder crear, editar o esborrar esdeveniments. D'aquesta forma, ens assegurem que els esdeveniments no vindran marcats segons l'interès polític, sinó que hi haurà una varietat per a tots els públics i ideologies.

#### 1.4.- Context

Actualment hi ha molt poques ciutats mitjanes que tinguin una aplicació mòbil i, les ciutats que en tenen, normalment consisteixen en aplicacions informatives sobre esdeveniments o informació rellevant sobre la ciutat. També hi ha ciutats amb aplicacions més simples, com per exemple Terrassa [4], que consisteix únicament en reportar incidències. Altres aplicacions, com per exemple la de Girona [5], si que són molt complertes. Tot i això, no està del tot segmentada ja que, malgrat que no és una ciutat gran com Barcelona o Madrid, també és una gran ciutat amb percentatge de turisme elevat i, per tant, l'aplicació no és vàlida per l'objectiu que busquem.

Per altra banda, ciutats grans com Barcelona, seu de l'*Smart City Expo World Congress*, estan creixent exponencialment amb el tema de ciutats intel·ligents i, tot i que sigui impossible competir-hi, és important que les ciutats mitjanes comencin a introduir-se al món de les tecnologies per no quedar-se enrere.

És per això que, analitzant el mercat actual, veiem que cada cop hi ha una diferència de creixement major entre les grans ciutats i les ciutats mitjanes i aquest projecte té l'objectiu de reduir, dins del seu abast, aquesta gran diferència. A més a més, cap de les aplicacions del mercat actual encaixa amb l'objectiu i característiques d'aquest projecte,

ja que les aplicacions actuals són massa genèriques, simples o poc segmentades i, tal com remarca *Malcolm McDonald* [6], professor d'*Oxford College of Marketing*, és molt important segmentar correctament els clients d'una empresa que, en aquest cas, fa referència als usuaris finals. És per això que aquesta aplicació pretén enfocar-se únicament als ciutadans per tal de facilitar-ne la comunicació amb l'ajuntament i els seus governants locals.

### 1.5.- Stakeholders

Un punt molt important en la gestió de projectes és identificar i descriure els *stakeholders*. Els *stakeholders* són els actors implicats en un projecte o empresa. En aquest cas, a qui va dirigit el projecte i qui participa en la planificació i desenvolupament. A continuació veurem els diferents rols definits tot i que els perfils relacionats amb el desenvolupament es duen a terme per una única persona.

- **Usuari:** usuari final de l'aplicació, el qual pot utilitzar i gaudir de tots els serveis oferts. Aquests usuaris són les persones residents a les ciutats mitjanes que d'alguna forma o una altra, volen participar, saber les últimes notícies i/o assistir a esdeveniments de la seva ciutat.
- **Administrador local:** organitzacions, entitats o clubs de la ciutat. Aquest usuari no va destinat a la aplicació mòbil. Tindrà una pàgina web d'administració on podrà penjar, modificar o editar esdeveniments que organitzin de forma que podran tenir un ressò a tota la ciutat.
- **Ajuntament:** persona o equip de persones encarregat de rebre i resoldre incidències; penjar, editar o esborrar esdeveniments; penjar enquestes per analitzar-ne el resultat a posteriori.
- **Dissenyador:** encarregat de dissenyar les diferents pantalles, el flux i el logotip de l'aplicació tenint en compte els criteris bàsics d'interfície d'usuari per aconseguir una bona experiència d'usuari.
- **Desenvolupador *front-end*:** encarregat de desenvolupar el software de l'aplicació Android basant-se amb els dissenys i flux elaborats pel dissenyador.

- **Desenvolupador *back-end*:** encarregat de desenvolupar els serveis *REST Api* necessaris per tal d'enllaçar la informació entre l'aplicació Android i la base de dades.
- **Desenvolupador web:** encarregat de desenvolupar la plataforma web per a l'administració de l'aplicació.
- **Administrador de bases de dades:** encarregar de crear, gestionar i mantenir les bases de dades que usará l'aplicació segons els criteris bàsics de seguretat, velocitat d'accés, concurrència i recuperació.
- **Project Manager:** encarregat de la gestió del projecte. El rol de *project manager* serveix per assignar les tasques durant diferents *sprints*. És un rol essencial per la correcta gestió d'un projecte.
- **Analista:** persona encarregada d'investigar quines són les tecnologies més òptimes pel projecte i analitzar les aplicacions de la competència.
- **Directora del projecte:** professora encarregada d'orientar, guiar i aconsellar al llarg del desenvolupament del projecte per tal d'aconseguir un bon desenvolupament i una bona execució del projecte, sempre dins dels terminis establerts. En aquest projecte, aquest rol el duu a terme la professora Silvia Llorente Viejo.

## 1.6.- Abast

Abans d'iniciar un projecte és important definir quin és l'abast del projecte i quins possibles impediments ens podem trobar al llarg del desenvolupament per tal de preveure fins on podem arribar i, a posteriori, veure fins on hem complert i poder obtenir un feedback d'aquest procés.

### 1.6.1.- Abast del projecte

En un projecte de desenvolupament mòbil hi ha cinc escenaris desenvolupats per diferents rols, tal i com hem vist a l'apartat 1.5. Aquest projecte pretén aconseguir el disseny i execució d'aquests cinc punts claus pel desenvolupament mòbil.

El procés més usual en el desenvolupament d'una aplicació és, en paral·lel, el desenvolupament mòbil, també conegut com a *front-end* i el desenvolupament del servidor *REST Api*, que es coneix com a *back-end*. Prèviament, és important tenir els dissenys de com volem aquests dos punts. Per tant, a priori, s'ha de fer el disseny de l'aplicació mòbil i el disseny de la base de dades.

Finalment, per aquest projecte és necessari el desenvolupament d'una pàgina web per tal de poder alliberar una mica més l'aplicació del partit polític que governi a la ciutat, de forma que les organitzacions d'ideologies diferents o lliures també puguin participar amb el calendari per poder crear una llista d'esdeveniments apolítica.

#### 1.6.2.- Possibles obstacles

En aquest punt es defineixen els possibles obstacles que poden sorgir al llarg del desenvolupament d'aquest projecte, tal i com és natural en qualsevol projecte tecnològic.

En primer lloc, un dels problemes més usuals és una mala estimació temporal de les tasques a dur a terme. És per això que en la planificació del projecte s'han de tenir en compte unes hores extres sense dedicació a cap tasca ja que, com és comú, l'estimació és difícil d'encertar en projectes tecnològics com aquest. A part, que una sola persona executi els diferents rols, augmenta el risc de mala estimació temporal ja que el desconeixement d'algunes àrees pot fer variar de forma pronunciada l'estimació. Tot i això, cal fer una estimació acurada i detallada i, sobretot, fer un seguiment constant de les tasques i de l'estimació per veure si progressa adequadament i per tal de reduir aquest risc al màxim.

En segon lloc, també pot ser que ens trobem limitacions físiques com podria ser una pèrdua de les dades del codi o de la base de dades a causa d'un dany a la memòria física. Això s'ha de tenir present per tal d'intentar utilitzar repositoris per guardar el codi i fer còpies de seguretat de la base de dades per tal de tenir un duplicat en cas de perdre l'original.

En tercer lloc, és possible que trobem dificultats a l'hora de muntar el servidor i fer-lo córrer. Donat que és una experiència nova, és probable que apareguin bastants

problemes i impediments que s'hauran de resoldre però poden portar temps i per tant, ampliar el temps d'execució.

Finalment, els obstacles personals que puguin sorgir, ja siguin malalties, problemes personals o manca d'hores per executar el treball. Aquest obstacle també s'ha de tenir en compte ja que no podem predir si hi haurà algun inconvenient personal que allargui el desenvolupament del projecte. És per això que també hem de tenir en compte que necessitem unes hores extres per cobrir qualsevol d'aquest problemes en cas que apareguin al llarg del desenvolupament.

### 1.7.- Rol personal del projecte

Tal i com s'indica, hi ha set rols principals per desenvolupar aquest projecte: dissenyador, desenvolupador *front-end*, desenvolupador *back-end*, desenvolupador web, administrador de bases de dades, analista i *project manager*.

Donat que, tal i com indica la normativa de Treballs de Finals de Grau de la Facultat d'Informàtica de Barcelona [7], aquest treball s'ha de desenvolupar de forma individual, els set rols seran assolits per la meua persona.

Això és un repte personal donat que el meu àmbit laboral és el desenvolupant *front-end* d'aplicacions mòbils i que mai he desenvolupat webs ni servidors excepte situacions puntuals per pràctiques o projectes universitaris. És per això que estic motivat per tal d'aprendre nous àmbits i sortir de la zona de confort habitual i tenir l'oportunitat d'aprendre nous llenguatges de programació o ampliar el coneixement dels que ja conec. També és important el rol de *project manager* per aprendre a gestionar totes les tasques i organitzar la càrrega de treball diària.

A més a més, és interessant el fet de poder plantejar i dissenyar personalment una aplicació mòbil, donat que no és una feina habitual a cap lloc de treball ja que, a menys que desenvolupis aplicacions pròpies, sempre hi ha un departament de disseny o el mateix client que faciliten el disseny.



## 2.- Estat de l'art

### 2.1.- Tecnologies existents

Abans de començar el desenvolupament, hem d'analitzar les diferents eines i tecnologies que ofereix el mercat, comparar-les analitzant els nostres requeriments i veure quina opció s'adapta més al nostre projecte i, per tant, quines acabem seleccionant.

#### 2.1.1.- Gestió de tasques

Per a la gestió de tasques hi ha diferents eines que permeten tenir d'una forma visual les tasques separades segons el seu estat de desenvolupament. Les més conegudes són *Jira* i *Trello*. *Jira* [8] és l'opció més completa ja que permet crear qualsevol estat que decidim per les tasques i cada tasca està definida amb un títol, descripció i comentaris. A més a més es pot fer un seguiment de les hores dedicades a cada tasca i veure de forma visual les hores dedicades a cada tasca. També ofereix la possibilitat de crear diferents *sprints* amb les seves tasques pertinents i afegir-hi l'estimació temporal. Tot i això, és una eina enfocada a empreses i per tant és de pagament.

Per altra banda, *Trello* [9] és una eina senzilla on es poden definir diferents blocs segons els diferents estats de les tasques i organitzar-ho d'una forma visual. Tot i ser molt més simple que *Jira*, és una eina *open source* i ofereix un ventall de possibilitats per organitzar les tasques per estats, tipus i etiquetes.

#### 2.1.2.- Gestió de codi

Per la gestió de codi s'utilitza el *Git* [10]. *Git* és un sistema de control de versions de codi on, similar a *Google Drive* o *Dropbox*, es pot guardar el codi. Tot i això, té els beneficis que el codi s'organitza de forma que és fàcil revertir errors i tenir el codi ben estructurat i organitzat.

Tot i que el *Git* es pot utilitzar per consola, hi ha softwares que ofereixen l'ús d'aquesta eina amb una interfície gràfica per facilitar-ne l'ús. Les dues eines més utilitzades i conegudes són *SourceTree* i *GitKraken*. *GitKraken* [11] és una eina molt fàcil d'utilitzar i

ofereix una interfície d'usuari bastant simple. Tot i això, només permet un període de temps gratuït. Per altra banda, tot i que *SourceTree* [12] no ofereix una interfície d'usuari tan fàcil d'entendre com *GitKraken*, ofereix les mateixes possibilitats i de forma gratuïta.

### 2.1.3.- Arquitectura de l'aplicació mòbil

Un punt molt rellevant en el desenvolupament mòbil és la seva arquitectura. És important perquè normalment el codi passa per mans de diferents programadors i, si el codi no està organitzat, entendre-ho pot arribar a ser molt complicat.

L'arquitectura més utilitzada i popular s'anomena *Clean Architecture* [13]. Aquesta arquitectura consisteix en separar el projecte en tres capes: capa de dades, on tindrem totes les classes d'objectes; capa de presentació, on trobarem tot el que faci referència a la interfície d'usuari i vistes; capa de domini, que és l'encarregada de la comunicació entre les primeres mencionades.

Utilitzant una arquitectura neta i organitzada com *Clean Architecture* es pot reutilitzar codi de forma senzilla ja que l'única capa que tindrà dependències d'*Android* serà la capa de presentació. Això implica que tota la capa de dades i domini pot ser reutilitzada per *iOS*.

També és important organitzar de forma clara la capa de presentació. Popularment hi ha tres patrons de disseny: *Model-View-Controller (MVC)*, *Model-View-Presenter (MVP)* i *Model-View-View-Model (MVVM)* [14].

- **MVC:** en aquest patró la *View* és l'encarregada de gestionar els esdeveniments i les interaccions de l'usuari. Els *Models* són els encarregats de comunicar-se amb la capa de domini per obtenir les dades. Finalment, el *Controller* és l'encarregat de comunicar als *Models* quan l'usuari interactua amb les *Views*.
- **MVP:** en aquest patró trobem tota la lògica relacionada amb la vista i interfície a la *View*. La *View* comunica tots els esdeveniments i interaccions de l'usuari al *Presenter*, que serà l'encarregat de gestionar la lògica. Aquest mateix *Presenter* serà l'encarregat de comunicar-se amb els *Models* per aconseguir les dades necessàries que s'han de mostrar a la *View*. La diferència principal amb MVC és

que cada *View* necessita un *Presenter*, de forma que conservem el codi molt més fragmentat i és molt més fàcil de mantenir.

- **MVVM**: en aquest patró tenim una *View* encarregada de gestionar tots els esdeveniments i interaccions de l'usuari i demanarà les dades al *View-Model*. Aquest *View-Model* s'encarrega de tota la lògica necessària per obtenir totes les dades que aconseguirà comunicant-se amb el *Model*. Bàsicament, el *View-Model* fa de pont entre la *View* i els *Models*.

S'utilitzi el patró que s'utilitzi és important que únicament les Views tinguin dependències d'*Android*. D'aquesta forma simplifiquem i facilitem el desenvolupament de *Tests Unitaris*.

#### 2.1.4.- Desenvolupament *front-end*

Pel desenvolupament d'aplicacions *Android*, Google ofereix un IDE (Integrated Development Environment) oficial, desenvolupat per l'empresa *IntelliJ IDEA* [15]. Aquest software ofereix un ampli ventall d'eines útils per desenvolupar aplicacions *Android*. Algunes d'aquestes eines són pel desenvolupament de tests, emuladors de mòbils per poder provar les aplicacions, integracions de *Git*, entre d'altres. A més a més, té suport de llibreries natives pel desenvolupament de les vistes i pantalles. També crea el projecte base amb l'estructura necessària per començar i suporta diferents llenguatges de programació, com ara C++, Kotlin o Java [16].

Tot i que es pot utilitzar C++, en el desenvolupament *Android* els llenguatges oficials i recomanats per Google són Java i Kotlin. La primera diferència entre aquests dos llenguatges és que Kotlin és un llenguatge “*null save*” per defecte. Això significa que amb Kotlin, per defecte, mai podrem tenir un error per intentar accedir a una posició que no existeix, anomenada *Null Point*. Aquest és un dels errors més habituals en la programació. Un segon punt important és que, tot i que Java també permet executar tasques en diferents *threads* per tal d'agilitzar les tasques, Kotlin ofereix les corutines. Les corutines són una forma molt fàcil i intuïtiva per executar diferents processos en paral·lel. En tercer lloc, Kotlin ofereix les anomenades “*data class*”. Aquest tipus de classes, faciliten la definició de classes que representen objectes ja que, per defecte, no

cal programar els *getters* i *setters*. Els *setters* ens permeten establir un valor als paràmetres de la classe i els *getters* obtenir-ne aquest valor. Actualment, un 66% de programadors Android utilitzen Kotlin com a una de les tres eines principals. [17]

Finalment, hi ha dues alternatives pel que fa a les bases de dades locals en *Android*. Aquestes dues alternatives són *Realm* i *Android Room* [18]. *Realm* és una llibreria per *Android* desenvolupada l'any 2014 que permet guardar dades en arxius locals al mòbil. Tot i que és cert que *Realm* permet major capacitat d'emmagatzematge respecte *Android Room*, aquest no permet l'accés paral·lel. Per altra banda, *Android Room*, desenvolupat oficialment per Google, és una llibreria nativa de *SQLite* i *Android*.

#### 2.1.5.- Desenvolupament back-end

Pel desenvolupament *back-end* es sol utilitzar PHP o JavaScript ja que són llenguatges simples i bastant complerts per dur a terme els serveis *REST Api*. També existeixen tecnologies més recents com TypeScript, basat en JavaScript, i SpringBoot.

JavaScript i PHP tenen molta documentació disponible donada la seva popularitat. TypeScript [19], al ser més recent, podria no tenir tanta documentació, però com que és un llenguatge basat en JavaScript, és bastant utilitzat i hi ha molts tutorials i documentació. En canvi, Spring Boot [20], tot i que permet un fàcil desenvolupament si ets un programador nou en *back-end*, té molt poca documentació i no és fàcil trobar tota la informació necessària.

Com a software de desenvolupament, hi ha varies opcions com NetBeans [21], VisualStudio [22] o IntelliJ Idea. VisualStudio i IntelliJ Idea permeten moltes facilitats i són bastant moderns i, com a conseqüència, tenen una interfície d'usuari més agradable.

#### 2.1.6.- Desenvolupament web

Els llenguatges més utilitzats pel desenvolupament web són JavaScript, PHP, Java i Python.

JavaScript [23] ofereix una bona integració amb HTML i CSS per poder desenvolupar el *front-end* de la web. És una llibreria molt utilitzada a nivell global i, per tant, ofereix moltes llibreries per facilitar-ne el desenvolupament.

Python [24] és el llenguatge més utilitzat pel desenvolupament web al món ja que és molt més senzill que les seves alternatives. Tot i que no ofereix un ventall tan ampli com JavaScript o Java, és molt fàcil d'aprendre i desenvolupar.

Tot i que Java és molt popular i ofereix moltes llibreries, està més enfocat a desenvolupament mòbil i, per tant, no està tant enfocat a web i no té gaire facilitats pel desenvolupament web.

PHP [25] no és de les eines més fàcils d'aprendre però té una gran oferta de possibilitats. Moltes de les grans webs, com per exemple *Wikipedia*, *Facebook* o *Yahoo* estan desenvolupades en PHP.

Igual que pel desenvolupament *back-end*, els softwares més utilitzats són *IntelliJ Idea* i *NetBeans*.

#### 2.1.7.- Administració de bases de dades

Hi ha dos tipus de bases de dades: les bases de dades relacionals i les no relacionals. Les bases de dades relacionals organitzen la informació en trossos petits que es relacionen entre ells amb identificadors. En canvi, a les bases de dades no relacionals la informació s'organitza en documents.

La principal diferència consisteix en que les bases de dades relacionals serveixen quan tenim un esquema clar i fix. Per contra, les bases de dades no relacionals, serveixen quan no tenim un esquema clar i fix de la informació que volem emmagatzemar.

Els exemples més populars són MySQL com a base de dades relacional i MongoDB com a base de dades no relacional [26].

A part, hi ha un gran ventall de softwares *open source* que permeten crear, editar, esborrar i administrar bases de dades. Per bases de dades relacionals, tenim un gran ventall d'alternatives, com MySQLWorkbench [27] o dbWatch [28] com a algunes de les més populars, però totes tenen unes funcionalitats i interfície molt semblants.

Per altra banda, per bases de dades no relacionals existeix MongoDB Compass [29]. Té una interfície molt usable i agradable i, a part, és molt fàcil d'utilitzar.

#### 2.1.8.- Documentació

Pel que fa a l'emmagatzematge de la documentació, les alternatives més utilitzades són *Google Drive* i *Dropbox* [30]. Les dues són molt semblants i ofereixen les mateixes prestacions.

#### 2.2.- Tecnologies seleccionades

Finalment, les tecnologies seleccionades són les següents:

- **Trello:** donat que *Jira* és de pagament, s'ha escollit *Trello* com a eina de gestió i organització de tasques.
- **SourceTree:** donat que, personalment, sóc més partidari d'utilitzar *GitKraken*, aquesta eina és, des de fa poc, de pagament i per tant, finalment, s'opta pel *SourceTree*.
- **Android Studio:** sense cap mena de dubte, com a software de desenvolupament de l'aplicació Android s'utilitza Android Studio ja que és l'oficial i el que ofereix més possibilitats i alternatives.
- **Kotlin:** com a llenguatge s'ha escollit Kotlin ja que, tot i ser cooficial amb Java, ofereix moltes més facilitats i és més segur pel que fa a errors de codi comparat amb Java.
- **Android Room:** tot i que *Realm* ofereix més capacitat que no pas *Android Room*, s'ha escollit *Android Room* ja que és l'eina oficial i permet fer accessos paral·lels.
- **Clean Architecture amb MVP:** per una preferència personal, escullo MVP donat que és l'arquitectura més neta i ordenada i més eficaç a l'hora d'identificar errors.

- **TypeScript:** donat que és una variant de JavaScript, té molta informació i documentació disponible a Internet. A part, a diferencia de JavaScript, la base de projecte és fàcil de desenvolupar i el manteniment és més senzill.
- **JavaScript+HTML+CSS:** pel desenvolupament web s'ha escollit aquesta combinació ja que és el llenguatge que més domino i és l'única de les alternatives que he treballat.
- **MongoDB:** donat que l'esquema de la base de dades pot anar variant i no hi ha una proposta fixa, s'escull bases de dades no relacionals. A més a més, hi ha una gran facilitat d'enllaçar MongoDB amb TypeScript, gràcies a la llibreria *mongoose*.
- **Visual Studio:** tot i que totes les alternatives són bastant semblants, aquest software és més modern i més senzill d'utilitzar que no pas *NetBeans* o *IntelliJ Idea*.
- **MongoDB Compass:** s'ha escollit aquesta eina donat que és la més popular per MongoDB i és molt usable.
- **Dropbox:** donat que ja està instal·lat i integrat al meu ordinador personal, he escollit Dropbox.

### 2.3.- Integració de coneixements

Els meus coneixements que puc posar en pràctica en aquest treball es diferencien en dos punts: els adquirits a la Universitat i els que he desenvolupat al món laboral.

Pel que fa a la Universitat, he après les bases de la programació i de desenvolupar algorismes. També he après diferents llenguatges de programació, cosa que m'ha facilitat sempre l'aprenentatge de nous llenguatges. Un punt molt important en aquest treball és l'administració de bases de dades que he desenvolupat al llarg de les diferents assignatures cursades a la Universitat. Finalment, els pocs coneixements que tinc de web els vaig adquirir a l'assignatura d'Aplicacions Distribuïdes que vaig cursar amb la directora d'aquest treball, Silvia Llorente, motiu pel qual vaig sol·licitar-li de dirigir aquest projecte.

Per altra banda, al llarg dels dos anys que porto al món laboral treballant com a desenvolupador Android he après a dominar el llenguatge de programació Kotlin i algunes de les seves llibreries més populars. Pel que fa a la gestió del codi, domino tant *GitKraken* com *SourceTree* donat que els he utilitzat tots dos a les diferents empreses on he treballat. Finalment, he après a gestionar i planificar les tasques amb metodologies àgils.



### 3.- Desenvolupament del projecte

Després de tot l'estudi sobre les ciutats mitjanes i la tecnologia i arribar a la conclusió de desenvolupar una aplicació mòbil per a aquestes, es va decidir fer l'aplicació sobre la meua ciutat, Badalona. És una ciutat que quadra perfectament amb totes les descripcions de ciutat mitjana ja que és una ciutat autosuficient, amb la seva economia pròpia resultat de la quantitat de comerços que hi ha, però amb poc turisme i un número d'habitants reduït comparat amb les grans ciutats.

A continuació veurem les cinc parts en que es divideix el desenvolupament del projecte: treball previ al desenvolupament, desenvolupament del servidor *back-end*, desenvolupament de la base de dades, desenvolupament de l'aplicació *Android* i desenvolupament web.

#### 3.1.- Treball previ al desenvolupament

Abans de començar el projecte, per saber com enfocar-lo i quins temes s'havien de tractar, vaig intentar posar-me en contacte amb els diferents líders dels diferents partits polítics de la meua ciutat, Badalona. Vaig aconseguir posar-me en contacte amb dos líders, el de Junts per Catalunya David Torrents i el del Partit Popular Xavier Garcia Albiol.

Després d'una llarga conversa amb David Torrents, on remarcava constantment la importància d'incloure tecnologies a la ciutat per tal de créixer, vam arribar a la conclusió que Badalona és una ciutat gran i autosuficient que està poc valorada i deixada a causa de la mala comunicació i la manca d'informació. Tot i que David Torrents en un primer moment volia enfocar el tema cap al turisme, li vaig comentar que aquest projecte va dirigir als habitants de la mateixa ciutat i no al turisme, ja que aquests són els que realment poden canviar el dia a dia de la ciutat. Un cop vam definir com havia de ser el projecte, em va comentar que, segons la seva opinió, l'ajuntament hauria de començar a invertir més en aconseguir una ciutat intel·ligent donat que facilitaria molt la vida dels seus ciutadans. Alguns punts que ell em comentava eren inviables donat que l'abast del meu projecte no era suficient, però tot i això vam arribar a unes conclusions finals:

- Molts ciutadans no saben què passa a la seva ciutat.
- Poca visibilitat en activitats locals.
- Es té poc en compte l'opinió dels ciutadans a l'hora de prendre decisions.
- Lentitud en arreglar o resoldre problemes i incidències a la via pública.

La xerrada amb Xavier Garcia Albiol va ser bastant més reduïda donat que em va comentar que era un tema que li interessava molt, però que només em concediria una entrevista en el cas que fos elegit alcalde de la ciutat. Com que finalment no ho va aconseguir, no es va interessar gaire més pel projecte. Tot i això, la breu xerrada amb el líder popular va ser interessant donat que el seu punt de vista també era que Badalona és una ciutat poc aprofitada i poc valorada i que això necessita un canvi. Aquest canvi és modernitzar la ciutat i aconseguir una ciutat intel·ligent per tal de facilitar el dia a dia dels ciutadans de Badalona.

Després de la xerrada amb dos líders de partits tan oposats i amb una opinió tant semblant, vaig definir les seccions de l'aplicació web:

- Notícies per difondre la informació de la ciutat.
- Llista d'esdeveniments per tal de donar visibilitat a activitats locals.
- Enquestes per facilitar la comunicació i donar veu als ciutadans.
- Incidències per tal d'oferir a l'ajuntament una via més ràpida de saber què s'ha d'arreglar a la via pública.

### 3.2.- Base de dades

La base de dades, tal i com es comenta a l'apartat de estat de l'art, és una base de dades no relacional creada amb l'eina MongoDB. Les taules que s'han creat són: usuaris, esdeveniments, incidències, enquestes i relació entre enquestes i usuaris.

La *figura 1* mostra el diagrama UML amb les taules creades i les seves relacions per tal d'entendre l'estructura interna de les dades.

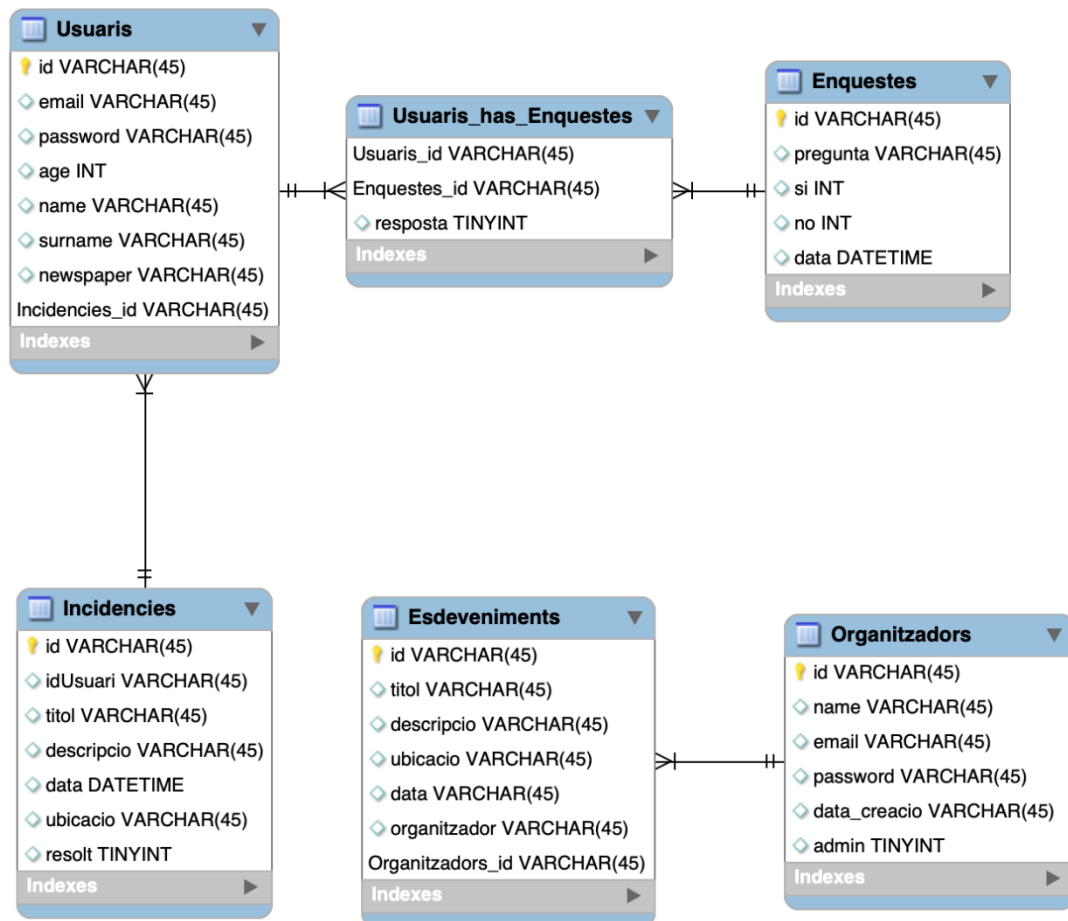


Figura 1. Diagrama UML de base de dades.

### 3.3.- Servidor back-end

El servidor back-end és l'encarregat de guardar i proporcionar les dades a l'aplicació. És per això que s'ha desenvolupat un servidor back-end que permet agafar i tornar fitxers de tipus JSON per tal de fer de túnel entre l'aplicació Android i la base de dades.

A continuació veurem l'arquitectura amb la que s'ha organitzat el projecte, les llibreries que s'han utilitzat i els diferents serveis que s'han desenvolupat.

#### 3.3.1.- Arquitectura

Per desenvolupar el servidor back-end s'ha utilitzat el llenguatge TypeScript i el software Visual Studio Code. Per organitzar el codi, s'ha utilitzat una estructura Model-View-Controller, d'aquesta forma, tenim el codi separat segons les rutes (*View*), lògica de

l'aplicació (*Controller*) i els models de dades (*Model*). A continuació, els directoris definits:

- **Config:** hi podem trobar tots els fitxers de configuració del servidor.
- **Controllers:** directori que conté els fitxers on es duu a terme la lògica de cada petició al servidor.
- **Middlewares:** conté els processos que s'executen abans de manejar les rutes del servei.
- **Models:** directori amb tots els models de dades que s'utilitzen i que, per tant, guardem a la base de dades.
- **Routes:** directori amb els fitxers on es descriuen les rutes amb la seva lògica associada. En aquest cas, tenim dos fitxers diferents: un per les rutes que no necessiten un *token* d'autenticació i un per les que sí que en necessiten.

Aquesta arquitectura serveix per tal de mantenir més ordenat el codi, de forma que qualsevol canvi no afecta a la resta del codi. A més a més, és molt més comprensible i, per tant, fàcil que algú ho pugui reutilitzar.

### 3.3.2.- Llibreries utilitzades

Tots els llenguatges existents tenen una sèrie de llibreries que faciliten el desenvolupament de codi. És per això, que en aquest punt s'anomenen i es descriuen les diferents llibreries utilitzades, per tal de veure perquè s'han utilitzat.

- **Dateformat:** llibreria que facilita la lectura i escriptura de dates. S'ha utilitzat per tal de guardar en el format correcte les dates de les incidències i del calendari.
- **Bcrypt:** llibreria que facilita l'enciptació de password. S'utilitza per tal de guardar les contrasenyes enciptades per mantenir la privacitat de l'usuari.
- **Express:** és el *framework* més popular de JavaScript per aquest tipus d'aplicacions. Ofereix moltes funcionalitats pel desenvolupament web, mòbil i *back-end*.

- **Jsonwebtoken:** és una llibreria que s'utilitza per facilitar l'autenticació per poder connectar-se als serveis.
- **Mongoose:** llibreria per connectar el servidor back-end amb la base de dades no relacional MongoDB.
- **Passport:** aquesta llibreria permet enllaçar la llibreria Jsonwebtoken amb les rutes on volem afegir el requisit d'autenticació per connectar-se.

### 3.3.3.- Serveis

A continuació veurem els diferents serveis que s'han desenvolupat en aquest projecte per tal de poder proveir l'aplicació amb les dades de la base de dades. Es descriu en els diferents serveis la seva ruta, que es mostra en el títol de cada secció; funcionalitat; *Request*; *Response* i possibles missatges d'error que poden enviar.

#### 3.3.3.1.- Registre (POST /signup)

Aquest servei serveix per registrar un usuari a l'aplicació. Per executar correctament el servei, l'aplicació ha d'enviar les dades necessàries pel registre: nom, cognoms, edat, correu electrònic, contrasenya i font de notícies per defecte escollit.

Aquest servei és l'encarregat d'encriptar la contrasenya de l'usuari abans de guardar-la a la base de dades. També és l'encarregat de generar un *token* per l'usuari per tal que pugui dur a terme correctament totes les peticions del sistema sense haver d'iniciar la sessió, ja que, amb el registre, s'inicia automàticament.

Si tot el procediment és correcte, es guarda a base de dades l'usuari amb tota la seva informació amb un identificador aleatori que crea el servidor.

Com que aquesta petició necessita rebre un JSON amb unes dades concretes pel correcte funcionament, és un mètode POST.

A continuació, es descriuen els paràmetres necessaris que s'han d'enviar a la petició de l'aplicació i a la seva resposta, juntament amb els *headers* necessaris:

**Headers:**

- “Content-Type: application/json”: Serveix per definir que enviem les dades en format JSON.

#### **Request:**

Les dades necessàries pel servei són:

- “name”: el nom de l’usuari. És de tipus *String*.
- “surname”: els cognoms de l’usuari. És de tipus *String*.
- “e-mail”: correu electrònic amb el que l’usuari es vol registrar. És de tipus *String*.
- “age”: edat de l’usuari. És de tipus *Number*.
- “password”: contrasenya de l’usuari. És de tipus *String*.
- “newspaper”: el nom del diari que vol que sigui la font d’informació. És de tipus *String*.

#### **Response:**

- “name”: el nom de l’usuari. És de tipus *String*.
- “surname”: els cognoms de l’usuari. És de tipus *String*.
- “e-mail”: correu electrònic amb el que l’usuari es vol registrar. És de tipus *String*.
- “age”: edat de l’usuari. És de tipus *Number*.
- “newspaper”: el nom del diari que vol que sigui la font d’informació. És de tipus *String*.
- “id”: identificador generat pel servidor. Serveix per definir l’id d’usuari. És de tipus *String*.
- “token”: token creat pel servei necessari per poder executar correctament els serveis privats d’usuari. Aquest *token* caduca al cap de 24 hores. És de tipus *String*.

Pel que fa als errors, és usual assignar un codi per cada possible error que pot produir que no es respongui amb les dades esperades. Els errors d’aquest servei són:

#### **Errors:**

- 401: no s’ha passat tota la informació necessària.

- 402: usuari ja existent.

### 3.3.3.2.- Iniciar Sessió (POST /signin)

Un usuari ja registrat, necessitarà iniciar la sessió a aplicació. Per poder executar aquest inici de sessió, l'usuari ha d'enviar el seu correu electrònic i la seva contrasenya. Aquesta contrasenya, és encriptada i comparada amb la contrasenya guardada anteriorment i, en el cas que les dades siguin correctes, genera i retorna un *token* per poder autenticar-se a l'hora d'executar les diverses funcionalitats que ofereix el sistema.

Com que es necessiten paràmetres a la petició, és de tipus POST. La petició i la resposta del servei és:

#### Headers:

- “Content-Type: application/json”: Serveix per definir que enviem les dades en format JSON.

#### Request:

Les dades necessàries pel servei són:

- “e-mail”: correu electrònic amb el que l'usuari s'ha registrat. És de tipus *String*.
- “password”: contrasenya de l'usuari. És de tipus *String*.

#### Response:

- “name”: el nom de l'usuari. És de tipus *String*.
- “surname”: els cognoms de l'usuari. És de tipus *String*.
- “e-mail”: correu electrònic amb el que l'usuari es vol registrar. És de tipus *String*.
- “age”: edat de l'usuari. És de tipus *Number*.
- “newspaper”: el nom del diari que vol que sigui la font d'informació. És de tipus *String*.
- “id”: id generat pel servidor. Serveix per definir l'identificador d'usuari. És de tipus *String*.

- “token”: *token* creat pel servei necessari per poder executar correctament els serveis privats d’usuari. Aquest *token* caduca al cap de 24 hores. És de tipus *String*.

Els errors que torna el servei en cas que les dades siguin incorrectes o no estiguin registrades a la base de dades són els següents:

**Errors:**

- 401: no s’ha passat tota la informació necessària.
- 402: usuari no existent.

*3.3.3.3.- Obtenir enquestes (POST /getPolls)*

S’obtenen totes les enquestes que l’ajuntament publica per tal de poder contestar. Ens retorna les enquestes i, en cas que l’usuari ja hagi contestat prèviament la pregunta, s’envia un paràmetre per identificar si es va respondre de forma afirmativa o negativa.

Per executar aquest servei és necessari que l’usuari estigui autenticat i tingui un *token* vàlid. És un servei de tipus POST, ja que necessita una petició amb paràmetres. La petició i la resposta és:

**Headers:**

- “Content-Type: application/json”: Serveix per definir que enviem les dades en format JSON.
- “Token”: *token* obtingut amb el registre o inici de sessió. Aquest *token* és privat de l’usuari i caduca al cap de 24 hores.

**Request:**

Les dades necessàries pel servei són:

- “user\_id”: identificador de l’usuari obtingut en el registre o inici de sessió. Serveix per identificar les respostes de l’usuari. És de tipus *String*.

**Response:**

- “poll\_id”: identificador generat pel servidor per cada pregunta. És de tipus *String*



- “question”: pregunta afegida per l’ajuntament que l’usuari ha de respondre. És de tipus *String*.
- “answer”: si l’usuari ha contestat prèviament aquesta pregunta, retorna *true* o *false* segons la resposta. És de tipus *Boolean*.

Aquest servei únicament té un possible error en el cas que no s’envii l’identificador d’usuari:

**Errors:**

- 401: no s’ha passat tota la informació necessària.

#### 3.3.3.4.- Respondre una enquesta (POST /answerPoll)

Aquest servei és l’encarregat de guardar a la base de dades la resposta de l’usuari per una pregunta concreta. Per poder identificar l’usuari i la pregunta, necessita els identificadors corresponents per poder registrar-ho a la base de dades. A més a més, necessita rebre una resposta en format *Boolean*, és a dir, amb dues úniques possibles respostes, *true* o *false*, segons si l’usuari respon amb una resposta afirmativa o negativa la pregunta.

Aquesta pregunta, en guardar-se, porta un comptador per cada pregunta per tal que l’administrador pugui saber quants vots positius i negatius té cada pregunta.

Aquest servei també necessita el *token* d’autenticació per poder executar-se i també és de tipus POST. La petició i la resposta és:

**Headers:**

- “Content-Type: application/json”: Serveix per definir que enviem les dades en format JSON.
- “Token”: *token* obtingut amb el registre o inici de sessió. Aquest *token* és privat de l’usuari i caduca al cap de 24 hores.

**Request:**

Les dades necessàries pel servei són:

- “user\_id”: identificador de l’usuari obtingut en el registre o inici de sessió. Serveix per identificar les respostes de l’usuari. És de tipus *String*.
- “poll\_id”: identificador de la pregunta obtinguda amb */getPolls*. Serveix per identificar la pregunta i poder guardar-ne la resposta de l’usuari. És de tipus *String*.
- “answer”: resposta de l’usuari. És de tipus *Boolean*.

**Response:** No hi ha resposta. Únicament el missatge del servidor.

Aquest servei únicament té un possible error en el cas que no s’envii l’identificador d’usuari:

**Errors:**

- 401: no s’ha passat tota la informació necessària.

### 3.3.3.5.- *Obtenir esdeveniments (GET /getEvents)*

Aquest servei proporciona tota la informació dels propers esdeveniments que es duen a terme a la ciutat. Aquest mètode és de tipus GET, ja que no cal que li passem cap informació, a part del *token* com a *Header*. Retorna el títol, descripció, ubicació, organitzador i data de l’esdeveniment.

**Headers:**

- “Content-Type: application/json”: Serveix per definir que enviem les dades en format JSON.
- “Token”: *token* obtingut amb el registre o inici de sessió. Aquest *token* és privat de l’usuari i caduca al cap de 24 hores.

**Request:** No es necessita cap paràmetre.

**Response:**

- “title”: títol de l’esdeveniment. És de tipus *String*.
- “description”: descripció de l’esdeveniment. És de tipus *String*.
- “date”: data i hora de l’esdeveniment. És de tipus *Date*.

- “location”: ubicació de l'esdeveniment. És de tipus *String*.
- “organizator”: organitzador de l'esdeveniment. És de tipus *String*.

**Errors:** no hi ha errors.

#### 3.3.3.6.- *Obtenir incidències (POST /getIncidents)*

Amb aquest servei s'obtenen les incidències que l'usuari ha reportat amb tota la seva informació: títol, descripció, data en que s'ha reportat la incidència i ubicació. També retorna si la incidència ha sigut resolta o no per tal de poder diferenciar-ho a l'aplicació.

Aquest servei també necessita el *token* d'autenticació per poder executar-se i també és de tipus POST. La petició i la resposta és:

##### **Headers:**

- “Content-Type: application/json”: Serveix per definir que enviem les dades en format JSON.
- “Token”: *token* obtingut amb el registre o inici de sessió. Aquest *token* és privat de l'usuari i caduca al cap de 24 hores.

##### **Request:**

Les dades necessàries pel servei són:

- “user\_id”: identificador de l'usuari obtingut en el registre o inici de sessió. Serveix per identificar les respostes de l'usuari. És de tipus *String*.

##### **Response:**

- “user\_id”: identificador de l'usuari obtingut en el registre o inici de sessió. Serveix per identificar les respostes de l'usuari. És de tipus *String*.
- “title”: títol de l'incidència. És de tipus *String*.
- “description”: descripció de l'incidència. És de tipus *String*.
- “date”: data i hora de l'incidència. És de tipus *Date*.
- “location”: ubicació de l'incidència. És de tipus *String*.
- “resolved”: marca si l'incidència ha sigut resolta o no. És de tipus *Boolean*.

Aquest servei únicament té un possible error en el cas que no s'envii l'identificador d'usuari:

#### **Errors:**

401: no s'ha passat tota la informació necessària.

### 3.4.- Aplicació Android

En aquest punt, igual que en l'anterior, s'explica l'arquitectura del projecte, les llibreries utilitzades per facilitar el desenvolupament i, finalment, el detall de totes les pantalles de l'aplicació, explicant els serveis utilitzats, els diferents comportaments i les possibles variacions.

#### 3.4.1.- Arquitectura

Tal i com s'explica a l'apartat d'estat de l'art, hi ha diverses formes per organitzar els fitxers per aplicacions Android. Per desenvolupar l'aplicació, s'ha escollit *Clean Architecture*. Aquesta arquitectura consisteix en tenir tres mòduls separats i independents entre ells:

- **data:** en aquest mòdul s'inclouen les llibreries i classes necessàries per la gestió de dades, ja sigui per obtenir-les des del servidor remot o des de base de dades.
- **domain:** aquest mòdul és el pont per passar les dades entre el mòdul de dades i el d'aplicació. Qualsevol petició que l'aplicació vulgui fer al mòdul de dades, passarà per aquest mòdul i a l'inrevés.
- **app:** aquest mòdul és l'encarregat de mostrar les vistes i tota la lògica de l'aplicació. Si necessita dades es comunica amb el mòdul de domini.

A més a més, el mòdul d'aplicació utilitza l'arquitectura *Model-View-Presenter*. Aquesta arquitectura consisteix en separar els mòduls segons les funcions, per tal que siguin independents entre ells:

- **Model:** són els models de dades que s'aconsegueixen a partir de la comunicació amb el mòdul de domini.

- **View:** encarregat de comunicar tots els esdeveniments i interaccions de l'usuari amb la interfície. També s'encarrega de mostrar tota la informació necessària a la pantalla.
- **Presenter:** encarregat de gestionar tota la lògica. També s'encarrega de comunicar-se amb els models per tal d'aconseguir les dades que s'han de mostrar a la vista.

### 3.4.2.- Llibreries utilitzades

En aquest punt es descriuen les diferents llibreries utilitzades per facilitar el desenvolupament de l'aplicació:

- **Kotlin:** llibreria utilitzada per poder utilitzar Kotlin enlloc de Java.
- **ConstraintLayout:** llibreria nativa d'*Android* que permet organitzar els elements de les pantalles de forma més ordenada i simple.
- **AppCompat:** llibreria per desenvolupar les pantalles de forma que sigui funcional tant per dispositius antics com els dispositius més recents.
- **CardView:** llibreria que aporta un element de disseny anomenat *Card*.
- **Glide:** llibreria per gestionar imatges a l'aplicació.
- **Kodein:** llibreria encarregada de la injecció de dependències. Treballar injectant dependències facilita la reutilització de codi, els *refactors* i el desenvolupament de tests.
- **RxJava:** llibreria encarregada de llegir les dades que rebem del servidor remot.
- **Twitter:** llibreria de *Twitter* per poder mostrar tuits a la nostra aplicació.
- **Mockito:** llibreria utilitzada per crear models i classes falses per poder desenvolupar els tests.

### 3.4.3.- Pantalles

#### 3.4.3.1- Splash Screen



*Figura 2. Captura de pantalla de l'Splash Screen*

La *figura 2* mostra el disseny de l'Splash Screen. A totes les aplicacions hi ha una pantalla inicial que apareix al llençar l'aplicació, on es comprova si hi ha una sessió iniciada o no, per tal de definir el primer flux de l'aplicació.

En aquesta pantalla es descarrega la informació necessària abans de que l'usuari pugui interactuar amb l'aplicació i es comprova si hi ha un usuari iniciat. En aquest cas, s'agafa de la memòria de l'aplicació, el *token* i l'identificador d'usuari que té guardats. Si aquestes dades estan guardades a memòria, s'accedeix a la pantalla *Home*. En cas que no estiguin guardades, s'accedeix a la pantalla d'Iniciar sessió per tal que l'usuari pugui iniciar sessió, en cas que tingui un e-mail registrat.

Aquesta pantalla no té cap interacció amb l'usuari, ja que únicament és una pantalla de descàrrega de dades.

#### 3.4.3.2.- Iniciar Sessió

Tal i com mostra la *figura 3*, en aquesta pantalla apareixen dos camps – correu electrònic i contrasenya –, un botó d'“Entrar” i, finalment, un text per usuaris no registrats.

Permet que l'usuari introdueixi les dades necessàries per poder iniciar sessió i poder

Aquesta és una captura de pantalla d'una interfície d'usuari per iniciar sessió. Al centre superior hi ha un logotip circular amb una 'B' blanca sobre un fons verd, amb el text 'Portal Badalona' just a sota. Sota el logotip hi ha dos camps de text: el primer està etiquetat amb un icona d'usuari i el text 'Nom d'usuari'; el segon està etiquetat amb un icona de clau i el text 'Contrasenya'. Just a sota del camp de contrasenya hi ha un text més petit que diu 'Has oblidat la teva contrasenya?'. Al centre de la pantalla hi ha un botó rectangular verd amb el text 'ENTRAR' en majúscules. Al peu de la pantalla hi ha un text que diu 'Encara no tens usuari? Registra't!'. La pantalla està envoltada d'un rectangle negre.

*Figura 3. Captura de pantalla d'iniciar sessió*

obtenir el *token* i identificador d'usuari necessaris per poder utilitzar totes les funcionalitats que veurem a posteriori.

The screenshot shows the login interface for 'Portal Badalona'. At the top is a logo with a stylized 'B' inside a teal circle. Below it, the text 'Portal Badalona' is displayed. There are two input fields: the first is labeled 'Nom d'usuari' with a red user icon, and the second is labeled 'Contrasenya' with a red key icon. Below the password field is a link that says 'Has oblidat la teva contrasenya?'. A red error message, 'Falta omplir algun camp', is centered on the page. Below this is a teal button labeled 'ENTRAR'. At the bottom, there is a link that says 'Encara no tens usuari? Registra't!'.

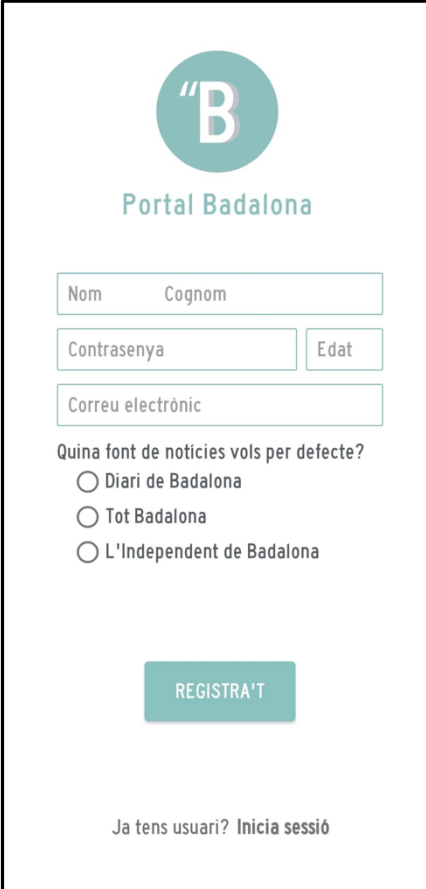
Figura 4. Captura de pantalla del registre erroni

El botó d'“Entrar” serveix per cridar el servei d'Iniciar Sessió. En cas que els dos camps necessaris estiguin omplerts, es crida el servei i es retorna la resposta. Si el servei respon amb un error, es mostra un missatge en vermell per tal que l'usuari sàpiga que no ha iniciat sessió correctament, tal i com indica la *figura 4*. En cas que l'usuari no hagi introduït alguna de les dades necessàries, es marca el requadre en vermell per identificar fàcilment el camp que falta i s'escriu un missatge.

Si es prem el botó de “Registra’t” es redirigeix a l'usuari a la pantalla de Registre.



### 3.4.3.3.- Registre



The screenshot shows a registration form for 'Portal Badalona'. At the top is a logo consisting of a teal circle with a white 'B' inside. Below the logo is the text 'Portal Badalona'. The form contains several input fields: a row for 'Nom' and 'Cognom', a row for 'Contrasenya' and 'Edat', and a single row for 'Correu electrònic'. Below these fields is a question 'Quina font de notícies vols per defecte?' followed by three radio button options: 'Diari de Badalona', 'Tot Badalona', and 'L'Independent de Badalona'. At the bottom of the form is a teal button labeled 'REGISTRA'T'. Below the button is a link that says 'Ja tens usuari? Inicia sessió'.

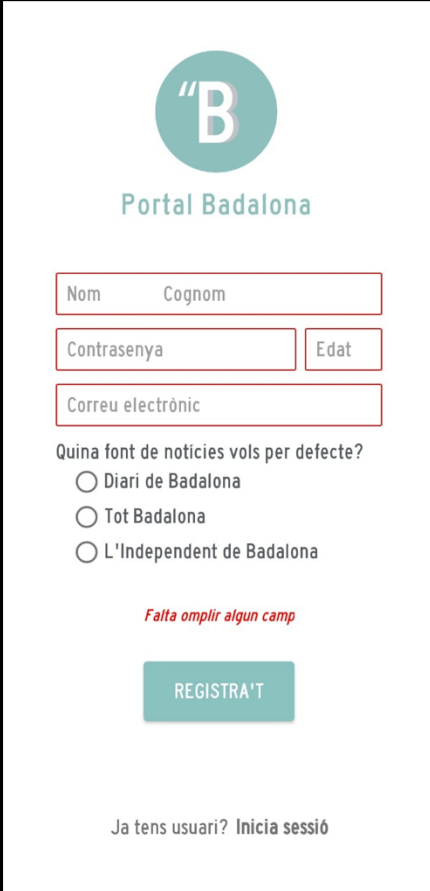
Figura 5. Captura de pantalla del registre

Com es pot veure a la *figura 5*, en aquesta pantalla apareixen diversos camps a omplir amb les dades de l'usuari. Aquestes dades introduïdes seran les necessàries per enviar al servei de Registre tots els paràmetres necessaris per poder registrar l'usuari correctament.

Ofereix diferents camps necessaris per poder registrar correctament un usuari: nom, cognom, edat, contrasenya, correu i la font de notícies que vol l'usuari per defecte.

Hi ha un botó de "Registrar" que, en el cas que totes les dades estiguin introduïdes, fa una crida al servei de registre i, si tot és correcte, es navega fins a la pàgina de *Home*. Si hi ha algun error al servei, tal i com podem veure a la *figura 6*, es mostra un missatge en vermell conforme hi ha hagut un error. En cas que falti algun camp, no es crida al servei i apareix un missatge conforme s'han d'omplir tots els camps.

En el cas de prémer “Iniciar Sessió”, es navega a la pàgina d’*Iniciar Sessió*.



The screenshot shows the 'Portal Badalona' registration page. At the top is a logo with a stylized 'B' inside a teal circle, followed by the text 'Portal Badalona'. Below this are several input fields: a combined field for 'Nom' and 'Cognom', a 'Contrasenya' field, an 'Edat' field, and a 'Correu electrònic' field. All these fields have red borders, indicating they contain errors. Below the fields is a question 'Quina font de notícies vols per defecte?' with three radio button options: 'Diari de Badalona', 'Tot Badalona', and 'L'Independent de Badalona'. A red error message 'Falta omplir algun camp' is displayed above a teal 'REGISTRA'T' button. At the bottom, there is a link 'Ja tens usuari? Inicia sessió'.

*Figura 6. Captura de pantalla del registre amb errors*

#### *3.4.3.4.- Menú*

Un cop la sessió s’ha iniciat, tenim un menú lateral on, cada cop que cliquem en un dels apartats que hi ha, es navega a aquella pantalla. Com es pot observar a la *figura 7*, les diferents opcions són: notícies, calendari, enquestes, incidències i configuració.

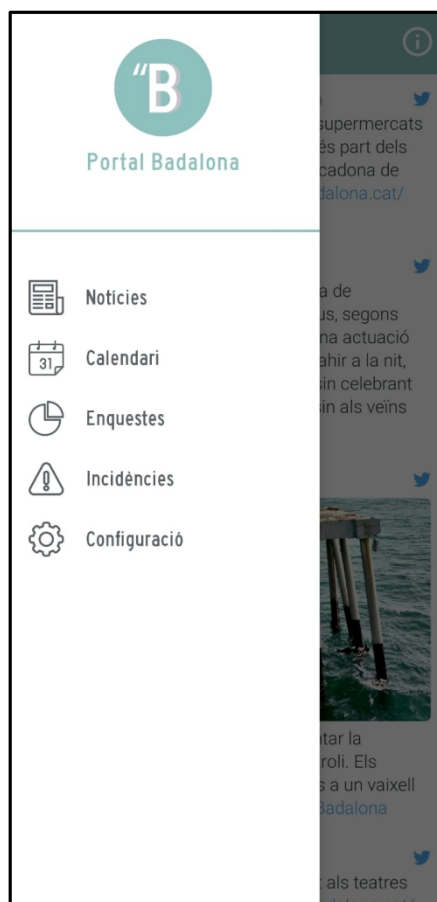


Figura 7. Captura de pantalla del menú

#### 3.4.3.5.- Notícies

Donat que cap dels diaris de Badalona ofereix cap forma d'aconseguir les notícies, s'ha optat per una alternativa. Finalment, s'utilitza l'API de *Twitter* per aconseguir els tuits que fan els diversos diaris de Badalona per tal de mostrar les notícies. Donat que l'API de *Twitter* només permet agafar els tuits d'un perfil a l'hora, la solució ha sigut que l'usuari triï quin vol que sigui el seu diari per defecte, de forma que sempre rebrà les notícies d'aquell diari. En cas que vulgui canviar-ho, haurà d'anar a Configuració.



Figura 8. Captura de pantalla de notícies

En resum, en aquesta pantalla veurem una llista de tuits del diari que hagi escollit l'usuari, tal i com mostra la *figura 8*. També trobem a la *toolbar* una icona d'informació. Si cliquem a la icona, apareix una alerta d'ajuda –com s'observa a la *figura 9*– on indica com canviar el diari per defecte. No utilitza cap servei propi del servidor, sinó que utilitza el servei de *Twitter*.



Figura 9. Captura de pantalla d'ajuda de notícies

#### 3.4.3.6.- Calendari

Tal i com es pot observar a la *figura 10*, es mostra una llista d'esdeveniments que l'ajuntament o organitzacions han afegit. Per obtenir els esdeveniments s'utilitza el servei d'obtenir esdeveniments. Cada esdeveniment mostra el títol de l'esdeveniment, la descripció, la ubicació de l'esdeveniment, la data i, finalment, l'organitzador que ha registrat l'esdeveniment.

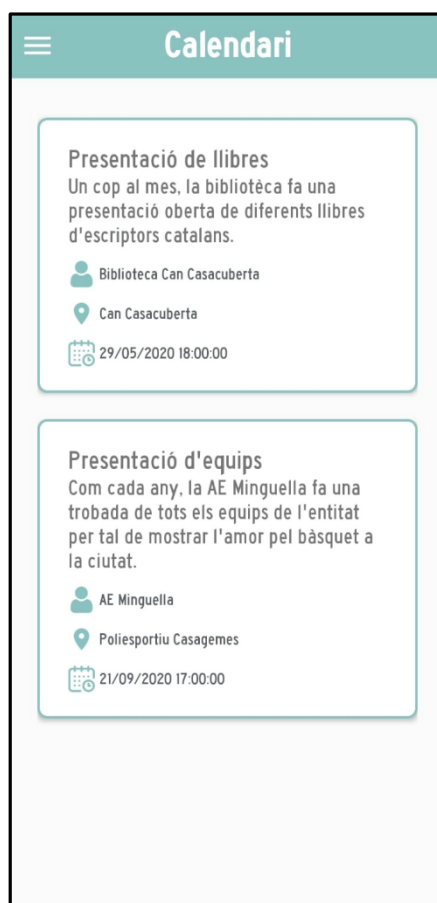


Figura 10. Captura de pantalla d'esdeveniments

#### 3.4.3.7.- Enquestes

La pantalla d'enquestes ofereix diverses preguntes que ha publicat l'ajuntament. S'utilitza el servei d'obtenir enquestes, de forma que retorna les enquestes amb la resposta que prèviament l'usuari havia marcat, en cas que ja hagués contestat.

Com es mostra a la *figura 11*, les preguntes ofereixen dos botons semblants als de "m'agrada" o "no m'agrada" de les xarxes socials. Quan seleccionem el que indica la resposta positiva, es marca el botó de color verd. En cas que seleccionem la resposta negativa, es marca de color vermell. Únicament pot haver-hi una resposta per pregunta, de forma que si tenim una resposta seleccionada i es selecciona l'oposada, la que estava seleccionada torna del color original i s'actualitza al servidor amb el servei d'actualitzar enquesta.

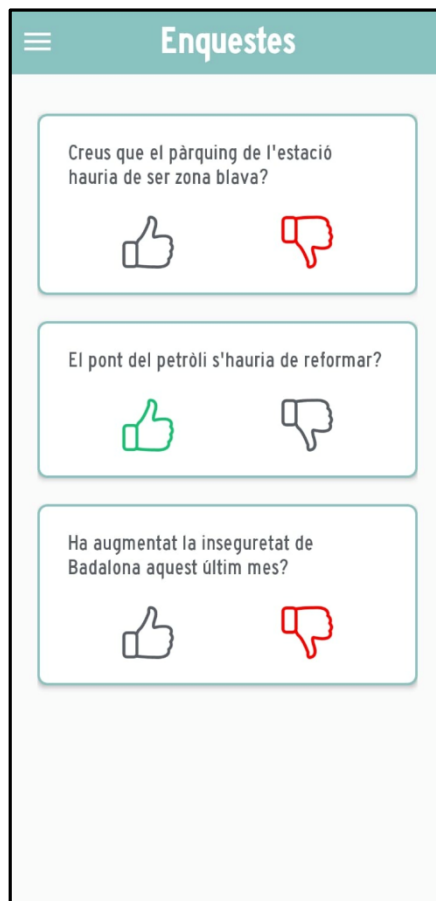


Figura 11. Captura de pantalla de les enquestes

#### 3.4.3.8.- Incidències

La pantalla mostra totes les incidències que hem reportat. Per tal d'obtenir-les, s'utilitza el servei d'obtenció d'incidències. En cas que l'ajuntament –és a dir, l'administrador– marqui la incidència com a resolta, es mostrarà el contorn de color verd per mostrar que ja s'ha resolt. En cas contrari, serà de color gris. A la figura 12 en veiem un exemple.



Figura 12. Captura de pantalla de les incidències

Per a cada incidència es mostra el títol, descripció, ubicació i data i hora del moment en que es registra la incidència. A més a més, es mostra visible un botó a la *toolbar* per poder navegar a la pantalla d'afegir una nova incidència.



### 3.4.3.9.- Nova incidència



← Nova Incidències

**Títol**

Banc trencat

**Descripció**

El banc de davant del McDonalds està trencat.

On es troba aquest problema?

Cantonada carrer del Mar i Rambla

AFEGIR

Figura 13. Captura de pantalla de nova incidència

Tal i com veiem a la *figura 13*, en aquesta pantalla trobem tres camps per omplir: títol, descripció i ubicació. En cas que intentem afegir la incidència i algun dels tres camps estigui buit, es mostra un error i s'indica amb color vermell quins són els camps erronis, com a la *figura 14*. En cas que es cliqui a la fletxa, es torna enrere i es mostra de nou la llista d'incidències. Un cop s'afegeix una incidència amb èxit, es torna a la pantalla de la llista d'incidències actualitzada per tal de mostrar la nova incidència.

← Nova Incidències

**Títol**

Banc trencat

**Descripció**

El banc de davant del McDonalds està trencat.

On es troba aquest problema?

Cantonada carrer del Mar i Rambla

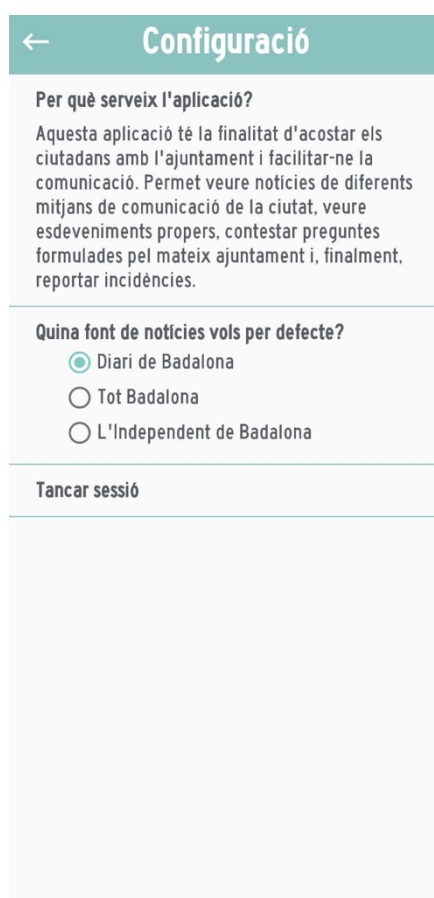
*Falta omplir algun camp*

AFEGIR

Figura 14. Captura de pantalla de nova incidència amb errors

Per afegir la incidència s'utilitza el servei d'afegir incidència. Aquest servei registra l'hora en que s'afegeix la incidència per tenir-ne constància.

### 3.4.3.10.- Configuració



← Configuració

**Per què serveix l'aplicació?**

Aquesta aplicació té la finalitat d'acostar els ciutadans amb l'ajuntament i facilitar-ne la comunicació. Permet veure notícies de diferents mitjans de comunicació de la ciutat, veure esdeveniments propers, contestar preguntes formulades pel mateix ajuntament i, finalment, reportar incidències.

**Quina font de notícies vols per defecte?**

☒ Diari de Badalona

☐ Tot Badalona

☐ L'Independent de Badalona

Tancar sessió

Figura 15. Captura de pantalla de configuració

Com podem veure a la *figura 15*, a la secció de configuració apareix un text explicatiu de les funcionalitats de l'aplicació. També trobem les diferents fonts de notícies incloses a l'aplicació per tal que l'usuari pugui escollir la que prefereixi. Finalment, tal i com mostra la *figura 16* ens permet tancar la sessió existent i tornar a la pantalla d'iniciar sessió.

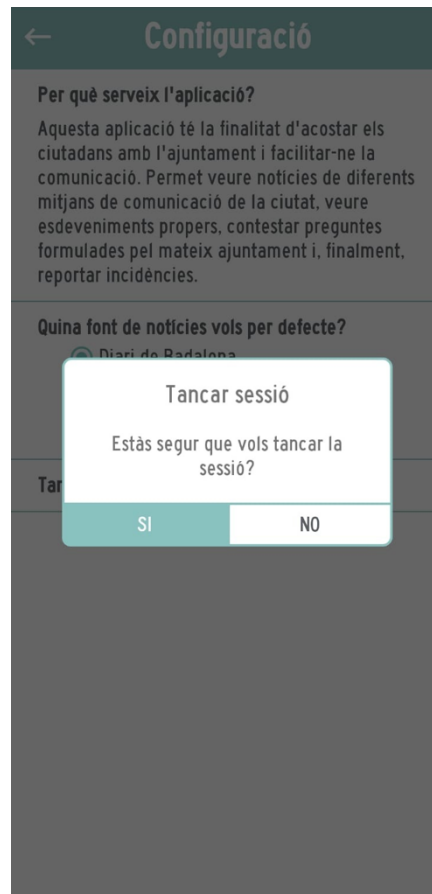


Figura 16. Captura de pantalla de tancar sessió

### 3.5.- Desenvolupament web

També s'ha hagut de desenvolupar una pàgina web per tal que l'Ajuntament pugui crear i editar esdeveniments, veure les incidències i canviar-ne l'estat a resolt, crear les enquestes i veure'n els resultats i, finalment, crear usuaris anomenats "organitzadors". També els anomenats "organitzadors" podran entrar a la pàgina web amb l'usuari i contrasenya que els han facilitat des de l'Ajuntament per tal de poder veure els seus esdeveniments o crear-ne de nous.

A continuació, com a les anteriors seccions, veurem quina arquitectura s'ha fet servir, quines són les llibreries que s'han utilitzat i, finalment, es comentaran les diferents pantalles de la pàgina web.

### 3.5.1.- Arquitectura

Pel que fa a l'arquitectura, es podria dir que s'ha utilitzat una arquitectura *Model-View-Controller*, ja que s'ha separat en diferents directoris els models, les vistes i els controladors. L'estructura del projecte és molt semblant a la del servidor:

- **Config:** aquesta carpeta conté els arxius de configuració de la pàgina web.
- **Helpers:** fitxer que defineixen funcions globals que es poden reutilitzar al llarg del projecte.
- **Models:** fitxers amb la definició de tots els models, és a dir, definir com aconseguim les dades de la base de dades.
- **Public:** carpeta amb els arxius estàtics com ara imatges o arxius CSS.
- **Routes:** es defineixen les rutes de la pàgina web.
- **Views:** emmagatzema els arxius que enviarem al navegador, és a dir, les vistes HTML.

Utilitzant una estructura tant simple com aquesta aconseguim tenir el codi ordenat i comprensible. A més a més, és molt més reutilitzable.

### 3.5.2.- Llibreries utilitzades

A continuació veurem les llibreries que s'utilitzen pel desenvolupament de la pàgina web:

- **Bcrypt:** llibreria que facilita l'enciptació de password. S'utilitza per tal de guardar les contrasenyes enciptades per mantenir la privacitat de l'usuari.
- **Connect-flash:** llibreria que ens permet mostrar missatges d'informació.
- **Express:** és el *framework* més popular de JavaScript per aquest tipus d'aplicacions. Ofereix moltes funcionalitats pel desenvolupament web, mòbil i *back-end*.
- **Mongoose:** llibreria per connectar el servidor back-end amb la base de dades no relacional MongoDB.

- **Passport:** aquesta llibreria permet enllaçar la llibreria Jsonwebtoken amb les rutes on volem afegir el requisit d'autenticació per connectar-se.

### 3.5.3.- Pantalles

Abans de veure la descripció de les pantalles, s'ha de descriure els dos tipus d'usuaris que tenim: els usuaris de la pàgina web els anomenem organitzadors, ja que la seva funció serà crear esdeveniments per tal que els diferents usuaris puguin veure esdeveniments de diferents entitats. Tot i això, l'Ajuntament ha de tenir un usuari d'organitzador administrador ja que el seu usuari ha de tenir més funcions que els altres.

#### 3.5.3.1.- Inici de sessió

La imatge mostra la interfície d'inici de sessió del Portal Badalona. A la part superior, hi ha una barra de navegació fosca amb el text "Portal Badalona" a l'esquerra i "Iniciar sessió" a la dreta. El fons principal és un gradient de verd clar a verd fosc. Al centre, hi ha un formulari blanc amb el títol "Iniciar sessió". Dins del formulari, hi ha un logotip circular amb una "B" blanca sobre un fons verd, seguit del text "Portal Badalona". A continuació, hi ha dos camps d'entrada: "Email" i "Contrasenya", amb botons de "Registar" a la dreta de cada camp. A la part inferior del formulari, hi ha un botó blau amb el text "Registar".

Figura 17. Captura de pantalla d'iniciar sessió (web)

A la *figura 17* trobem un formulari amb dos camps: correu electrònic i contrasenya. En cas que introduïm dades errònies tenim dos possibles missatges. El primer, d'usuari inexistent, tal i com podem veure a la *figura 18* i, en segon lloc, contrasenya incorrecta tal i com podem veure a la *figura 19*.

La imatge mostra un missatge d'error d'usuari inexistent. El missatge és un rectangle horitzontal de color rosa clar amb una franja superior de color verd clar. Dins del rectangle, hi ha el text "L'usuari no existeix" a l'esquerra i un icona de "x" a la dreta.

Figura 18. Captura de pantalla del missatge d'error d'usuari inexistent (web)

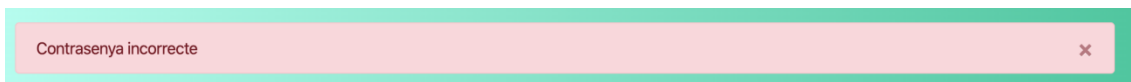


Figura 19. Captura de pantalla el missatge d'error de contrasenya incorrecta

En cas que introduïm les dades correctes ens dirigeix a la pantalla de llista d'esdeveniments.

### 3.5.3.2.- Llista d'esdeveniments

A screenshot of a web application interface. At the top is a dark navigation bar with the text "Portal Badalona" and several dropdown menus: "Esdeveniments", "Incidències", "Organitzadors", and "Enquestes". On the right of this bar is a link "Tancar sessió". Below the navigation bar is a table with a light green background. The table has five columns: "Títol", "Descripció", "Ubicació", "Data", and "Acció". There are three rows of event data, each with a blue "Editar" button and a red "Esborrar" button in the "Acció" column.

Títol	Descripció	Ubicació	Data	Acció
Presentació d'equips	Com cada any, l'AE Minguella fa desfilar tots els seus equips per tal de mostrar a la ciutat de Badalona la gran feina del club.	Poliesportiu Casagemes	Wed Apr 15 2020 15:00:00 GMT+0200 (GMT+02:00)	<button>Editar</button> <button>Esborrar</button>
Remember	La millor festa de Setmana Santa. T'hi esperem!	Titus Carpa	Sun Apr 12 2020 12:47:28 GMT+0200 (GMT+02:00)	<button>Editar</button> <button>Esborrar</button>
Presentació d'un llibre	Presentem un llibre	Biblioteca Can Casacuberta	Sun Apr 12 2020 12:41:59 GMT+0200 (GMT+02:00)	<button>Editar</button> <button>Esborrar</button>

Figura 20. Captura de pantalla de llista d'esdeveniments (web)

Tal i com veiem a la *figura 20*, apareix una *llista* dels esdeveniments creats per l'usuari. En aquesta figura es mostra la barra de navegació de l'usuari administrador. En cas que sigui un organitzador normal i corrent, apareix la barra de navegació com a la *figura 21*.



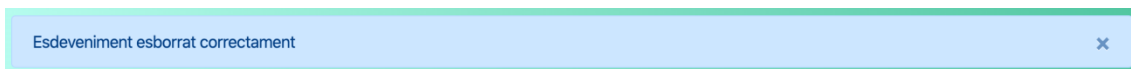
Figura 21. Captura de pantalla de la barra de navegació d'organitzador no administrador (web)

Cada esdeveniment tindrà un títol, descripció, ubicació i la data. També tindrà dos botons, un per eliminar l'esdeveniment i un per editar-lo.

En cas d'editar-lo correctament, apareix un missatge com indica la *figura 22*. En cas que s'esborri un esdeveniment, es mostra el missatge de la *figura 23*.

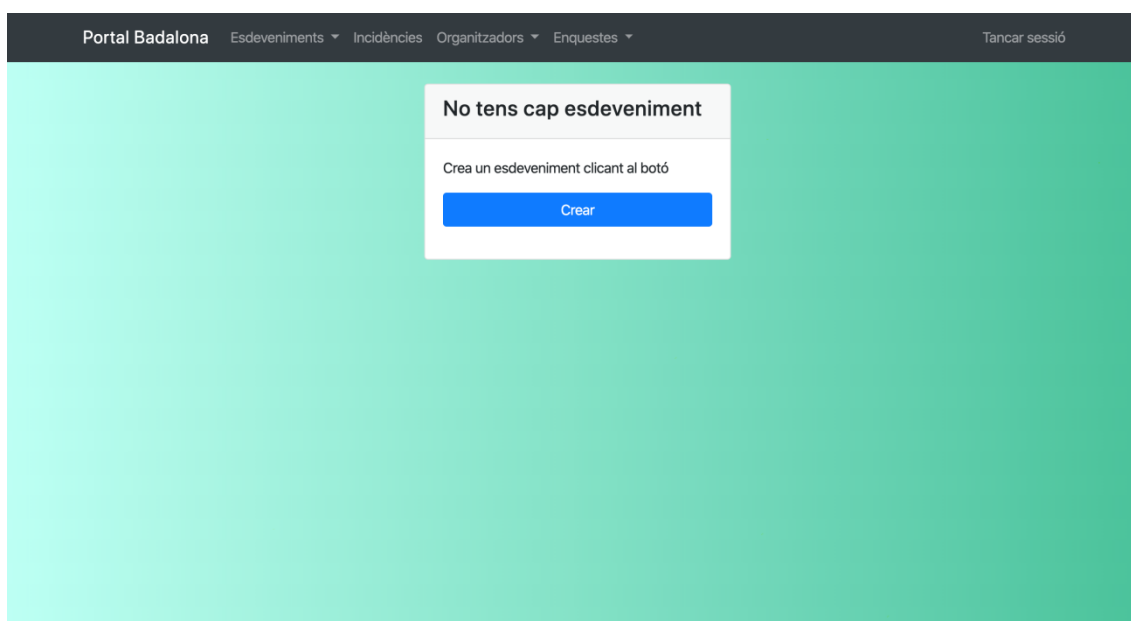


*Figura 22. Captura de pantalla del missatge d'esdeveniment actualitzat (web)*



*Figura 23. Captura de pantalla del missatge d'esdeveniment esborrat (web)*

En cas que aquest usuari no tingui cap esdeveniment creat, tal i com mostra la *figura 24*, veurà un missatge amb un botó que el redirigirà a la pàgina d'afegir esdeveniments.



*Figura 24. Captura de pantalla de la llista d'esdeveniments quan és buida (web)*



### 3.5.3.3.- Editar esdeveniment

The screenshot shows the 'Editar esdeveniment' form in the Portal Badalona system. The form is titled 'Editar esdeveniment' and contains the following fields:

- Títol:** Presentació d'un llibre
- Descripció:** Com fa la biblioteca cada divendres, es presentarà un llibre a la biblioteca
- Ubicació:** Biblioteca Can Casacuberta
- Data:** 13/06/2020 17:30

A blue 'Guardar' button is located at the bottom of the form.

Figura 25. Captura de pantalla d'editar esdeveniment (web)

Tal i com mostra la *figura 25*, es pot veure la pantalla d'editar un esdeveniment. En aquest cas, es mostra un formulari completat amb les dades actuals de l'esdeveniment que es poden modificar.

### 3.5.3.4.- Afegir esdeveniment

The screenshot shows the 'Nou esdeveniment' form in the Portal Badalona system. The form is titled 'Nou esdeveniment' and contains the following fields:

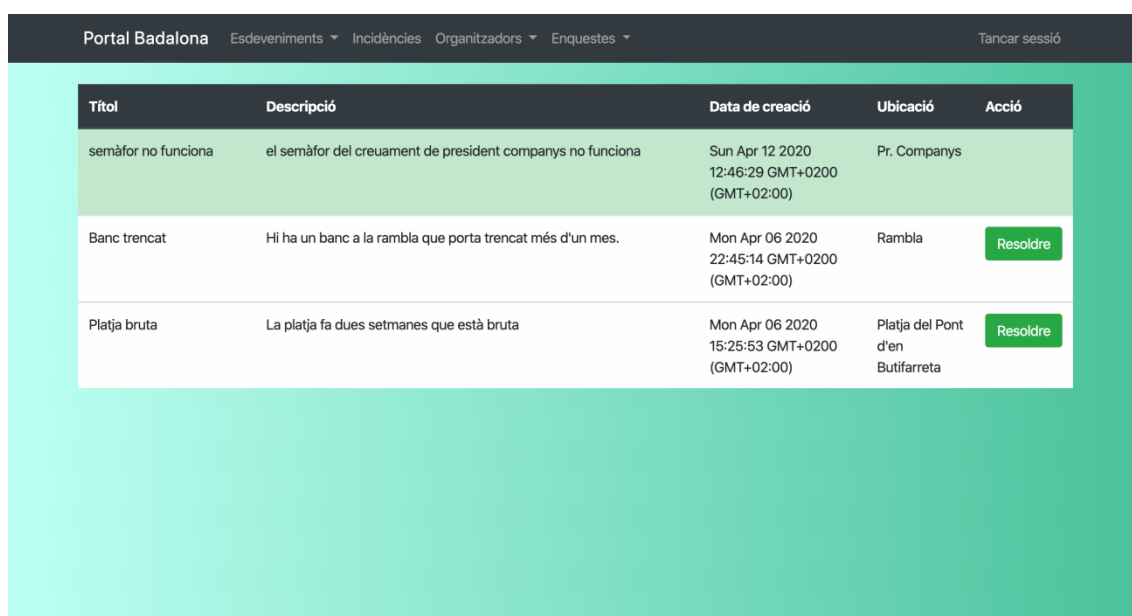
- Títol:** Títol esdeveniment
- Descripció:** Descripció de l'esdeveniment
- Ubicació:** Ubicació
- Data:** dd/mm/aaaa --:--

A blue 'Guardar' button is located at the bottom of the form.

Figura 26. Captura de pantalla d'afegir un esdeveniment

Com mostra la *figura 26*, en cas que es vulgui afegir un esdeveniment apareix un formulari amb els següents camps: títol, descripció, ubicació i data. En cas que cliquem el botó de guardar, es redirigeix a la pantalla de llista d'esdeveniments.

### 3.5.3.5.- Incidències

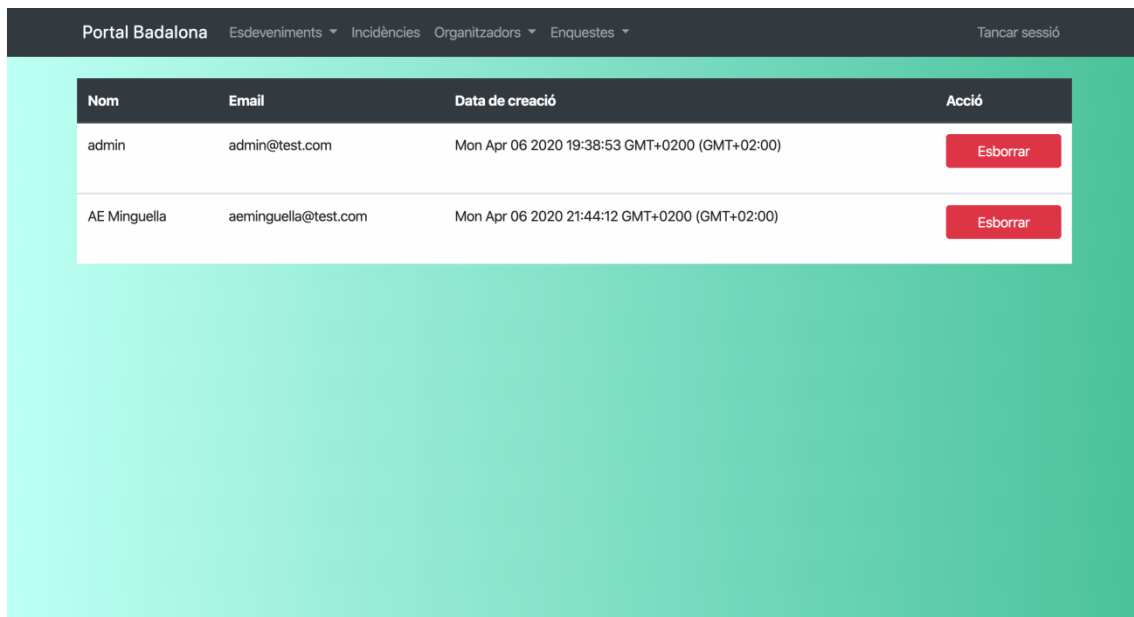


Títol	Descripció	Data de creació	Ubicació	Acció
semàfor no funciona	el semàfor del creuament de president companys no funciona	Sun Apr 12 2020 12:46:29 GMT+0200 (GMT+02:00)	Pr. Companys	
Banc trencat	Hi ha un banc a la rambla que porta trencat més d'un mes.	Mon Apr 06 2020 22:45:14 GMT+0200 (GMT+02:00)	Rambla	Resoldre
Platja bruta	La platja fa dues setmanes que està bruta	Mon Apr 06 2020 15:25:53 GMT+0200 (GMT+02:00)	Platja del Pont d'en Butifarreta	Resoldre

*Figura 27. Captura de pantalla de llista d'incidències*

Tal i com mostra la *figura 27*, en cas que siguem organitzadors administradors, podrem veure una llista d'incidències reportades pels usuaris de l'aplicació mòbil. En aquest cas, cada incidència mostra el títol, descripció, data de creació i ubicació. També apareix un botó per resoldre la incidència. En cas que es marqui com a resolta, es marca de color verd.

### 3.5.3.6.- Llista d'organitzadors



Nom	Email	Data de creació	Acció
admin	admin@test.com	Mon Apr 06 2020 19:38:53 GMT+0200 (GMT+02:00)	<button>Esborrar</button>
AE Minguella	aeminguella@test.com	Mon Apr 06 2020 21:44:12 GMT+0200 (GMT+02:00)	<button>Esborrar</button>

Figura 28. Captura de pantalla de llista d'organitzadors (web)

A la *figura 28* es pot veure la llista d'organitzadors. Aquesta pàgina existeix ja que convé que els organitzadors estiguin validats per l'Ajuntament, de forma que no es pot registrar un usuari qualsevol, sinó que l'Ajuntament ha d'aprovar la creació d'aquests perfils. Cada organitzador tindrà un nom, correu electrònic i data de creació. També tindrà un botó per esborrar el perfil escollit.

En cas que esborrem un organitzador, es mostra el missatge de la *figura 29*. En cas que un organitzador sigui afegir correctament, es mostra el missatge de la *figura 30*.



Figura 29. Captura de pantalla del missatge d'organitzador registrat (web)

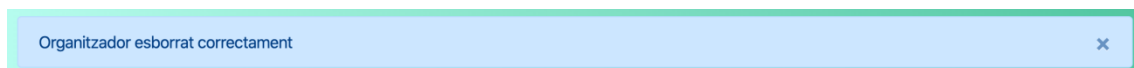


Figura 30. Captura de pantalla del missatge d'organitzador esborrat (web)

### 3.5.3.7.- Afegir organitzador

The screenshot shows a web form titled "Registrar una organització o entitat" (Register an organization or entity). The form is set against a light green background. It contains four input fields: "Nom de l'entitat" (Entity name), "Email", "Contrasenya" (Password), and "Repetir contrasenya" (Repeat password). Below these fields is a blue button labeled "Registrar". The top navigation bar includes "Portal Badalona" and several menu items: "Esdeveniments", "Incidències", "Organitzadors", "Enquestes", and "Tancar sessió".

Figura 31. Captura de pantalla d'afegir un organitzador (web)

A la figura 31 es mostra el formulari amb les dades necessàries per poder registrar un organitzador. En aquest cas, l'Ajuntament escriuria la contrasenya i, a posteriori, l'enviaria a l'organització o entitat. En cas que el formulari es registri correctament, es redirigirà a la pantalla de llista d'organitzadors.

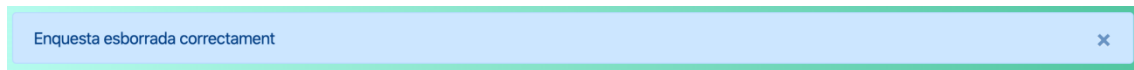
### 3.5.3.8.- Llista d'enquestes

The screenshot shows a table titled "Llista d'enquestes" (List of surveys). The table has five columns: "Pregunta" (Question), "Respostes positives" (Positive responses), "Respostes negatives" (Negative responses), "Data de creació" (Creation date), and "Acció" (Action). There are two rows of survey data. The first row is for the question "El Sant Josep hauria de tancar les seves instal·lacions?" (St. Joseph should close its facilities?), with 0 positive and 0 negative responses, created on Sun Apr 12 2020 21:02:01 GMT+0200. The second row is for "S'hauria de reformar el pont del petroli?" (Should the oil bridge be reformed?), also with 0 positive and 0 negative responses, created on Sun Apr 12 2020 21:01:35 GMT+0200. Each row has a red "Esborrar" (Delete) button in the "Acció" column. The top navigation bar is identical to the previous screenshot.

Pregunta	Respostes positives	Respostes negatives	Data de creació	Acció
El Sant Josep hauria de tancar les seves instal·lacions?	0	0	Sun Apr 12 2020 21:02:01 GMT+0200 (GMT+02:00)	<a href="#">Esborrar</a>
S'hauria de reformar el pont del petroli?	0	0	Sun Apr 12 2020 21:01:35 GMT+0200 (GMT+02:00)	<a href="#">Esborrar</a>

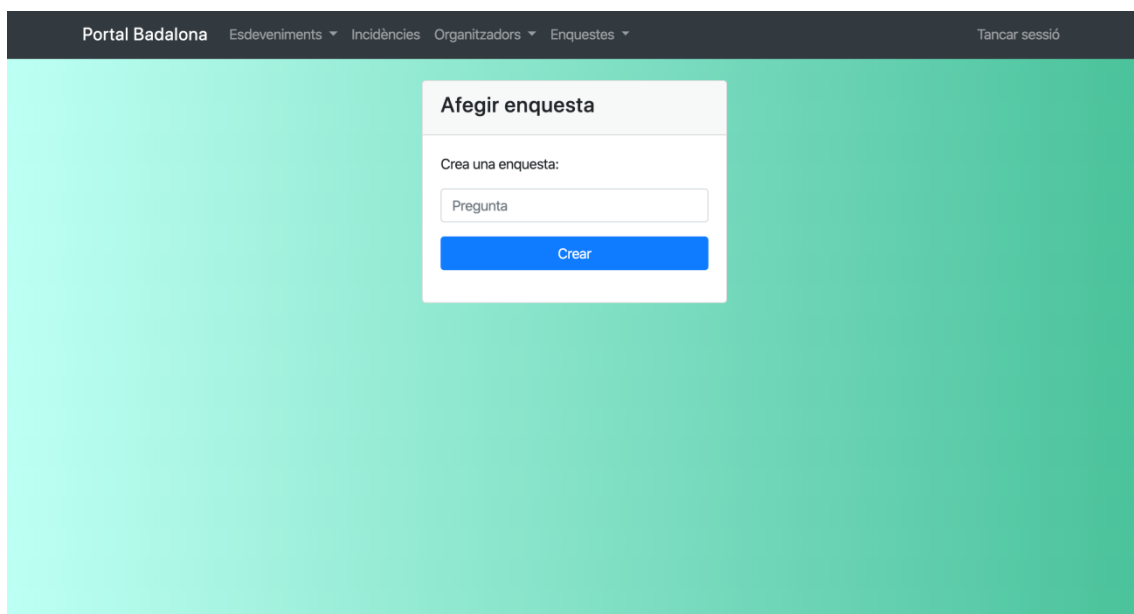
Figura 32. Captura de pantalla de llista d'enquestes (web)

Com es pot veure a la *figura 32*, apareix una llista amb la pregunta, el nombre de respostes positives i negatives per part de l'ajuntament i la seva data de creació. Cada enquesta també tindrà un botó d'esborrar. En cas que s'esborri correctament una enquesta, es mostrarà el missatge de la *figura 33*.



*Figura 33. Captura de pantalla del missatge d'enquesta esborrada (web)*

En cas que no hi hagi cap enquesta creada, es veurà un missatge i un camp per poder afegir una pregunta, tal i com es mostra a la *figura 34*.



*Figura 34. Captura de pantalla de llista d'enquestes buida (web)*

Quan s'afegeix una enquesta, es redirigeix a la pantalla de llista d'enquestes i es mostra el missatge de la *figura 35*.



*Figura 35. Captura de pantalla del missatge d'enquesta creada (web)*

#### *3.5.3.9.- Afegir enquesta*

Finalment, hi ha la pantalla d'afegir enquesta. La vista és igual que la *figura 34*, tal i com es mostra la pantalla de llista d'enquestes quan està buida.

### 3.6. Repositoris del projecte

Tot el codi desenvolupat per aquest projecte està en tres repositoris diferents. A continuació, es mostren els enllaços per poder accedir al codi:

- Codi de l'aplicació: <https://github.com/jordirubiralta/portalbdn>
- Codi de la pàgina web: <https://github.com/jordirubiralta/portalBdnWeb>
- Codi del servidor: <https://github.com/jordirubiralta/portalbdnApi>

## 4.- Metodologia

Totes les eines esmentades en aquest apartat han estat comparades amb les seves alternatives. Aquestes comparatives i decisions les podem trobar a l'apartat estat de l'art.

Donat que aquest projecte no té uns requisits i objectius tancats, sinó que està obert a canvis i adaptacions al llarg del desenvolupament, les metodologies àgils són les que més s'adapten a les nostres necessitats. Les metodologies àgils es basen en minimitzar el risc d'un projecte dividint-lo en petits projectes d'una durada aproximada d'entre una i quatre setmanes anomenats *sprints*. Cada *sprint* acaba amb una part del projecte dissenyada, desenvolupada i testejada.

Segons el *Chaos Manifesto* [31] els petits projectes tenen un 68% de possibilitats d'èxit i, per contra, els projectes grans únicament un 6%. Per tant, el que es planteja en el *Chaos Manifesto* és fragmentar un gran projecte en petits projectes per tal de tenir més possibilitats d'èxit i, això, és el que es busca amb les metodologies àgils. Aquestes metodologies estan preparades per adaptar-se a qualsevol canvi necessari per tal que el projecte sigui exitós.

### 4.1.- Organització de tasques

Per tal de poder dividir el projecte tal i com s'ha plantejat i fer un seguiment de les tasques, es farà ús de l'eina *Trello*. Aquesta eina és una web gratuïta que s'utilitza per gestionar projectes de manera que es poden organitzar les tasques per blocs per a cada *sprint*. A més a més, les tasques es diferenciarien segons el seu estat per una millor organització:

- **TO DO:** tasques descrites que s'han de desenvolupar a l'*sprint* actual.
- **IN PROGRESS:** tasques que estan sent desenvolupades en aquest moment pel desenvolupador.
- **CODE REVIEW:** tasques pendents de revisió de codi per tal d'assegurar-se que el codi és correcte. En un principi aquest procés el fan desenvolupadors diferents per tal de poder aportar coneixements i idees diferents i donar feedback a qui ha

dut a terme la tasca. Tot i això, en aquest cas, serà el mateix desenvolupador que farà una pròpia revisió de codi per evitar errors.

- **QA:** tasques preparades per ser testejades i provar que totes les funcionalitats funcionen correctament.
- **DONE:** tasques acabades correctament que ja han sigut provades i testejades.

És important fer un bon seguiment i posar les tasques en el seu estat correcte per poder seguir la planificació temporal i anar veient si hi ha desviacions o si la planificació estimada és correcta.

#### 4.2.- Organització de codi

Per la gestió de codi s'utilitza una eina anomenada *Git-Flow*. Aquesta metodologia es basa en separar el codi en branques per tenir sempre el codi organitzat. D'aquesta forma sempre tindrem les tasques separades de forma que qualsevol error es pot desfer fàcilment sense perjudicar el codi que ja està provat i testejat. Aquesta metodologia està organitzada en diferents branques segons la seva funció:

- **Master:** branca amb tot el codi que hi ha està preparat per pujar a producció, en aquest cas a *Google Play*.
- **Develop:** branca amb tot el codi per la següent versió planificada del projecte.
- **Feature:** branca que neix de *develop* i serveix per desenvolupar qualsevol nova tasca.
- **Release:** branca que neix de *develop* i serveix per generar una nova versió de producció. Quan es tanca aquesta branca, es porta el codi a la branca *master*.
- **Hotfix:** branca que neix de *master* i serveix per corregir un *bug* del codi de producció.

Per facilitar la gestió de la metodologia *Git-Flow* hi ha dos programes molt populars que són els més utilitzats pels desenvolupadors. Nosaltres hem escollit l'eina *SourceTree*.



Per la documentació s'ha escollit *Dropbox*. És una eina gratuïta que permet guardar tot tipus de documents al núvol. És fàcil d'integrar a l'ordinador, té un accés força ràpid i permet força capacitat d'emmagatzematge.

Pel desenvolupament del codi s'utilitza *Android Studio* i *IntelliJ Idea* per l'aplicació mòbil i pels serveis, respectivament. Són dues eines gratuïtes de desenvolupament mòbil, webs i serveis.

#### 4.3.- Gestió de bases de dades

Per a la gestió de les bases de dades utilitzarem *DB Browser for SQLite*. És una eina *open source* que permet crear, editar i esborrar bases de dades ja sigui de forma visual o amb codi SQL. És molt senzilla d'utilitzar i ofereix un gran ventall de facilitats.

## 5.- Planificació temporal

A continuació, veurem les tasques que s'han de dur a terme per desenvolupar el projecte. Les tasques seran agrupades segons el seu caràcter, tenint en compte els quatre objectius descrits a l'apartat 1.3. També es tindran en compte les tasques relacionades amb la gestió del projecte.

L'inici del Treball és al Setembre i la data final al Gener. Tot i això, en cas que es patís una desviació, l'entrega podria allargar-se fins a finals d'Abril.

### 5.1.- Fases del projecte

El projecte es divideix en cinc fases: desenvolupament Android, desenvolupament de bases de dades, desenvolupament de la pàgina web, desenvolupament dels serveis REST i gestió del projecte.

#### 5.1.1.- Desenvolupament Android

Per dur a terme aquestes tasques, els únics recursos necessaris són l'ordinador per desenvolupar el codi i poder executar-lo i un dispositiu Android per poder provar l'aplicació i facilitar la feina. Tot i això, el dispositiu Android es podria substituir per un emulador que ofereix el mateix Android Studio.

**T1.1.** Disseny del flux i *Activities* de l'aplicació mòbil. **(Temps estimat: 8 hores)**

**T1.2.** Desenvolupament del projecte base *Android*. Abans de començar a desenvolupar, és important organitzar el projecte segons l'arquitectura *Clean Architecture* descrita al punt d'estat de l'art relacionat amb l'arquitectura. També cal afegir totes les llibreries necessàries pel desenvolupament del projecte. **(Temps estimat: 10 hores)**

**T1.3.** Desenvolupament de la pantalla *Login*. Creació dels *layouts* i integrar les funcionalitats i serveis relacionats amb l'inici de sessió. **(Temps estimat: 10 hores)**

**T1.4.** Desenvolupament de la pantalla Registre. Creació dels *layouts* i integrar les funcionalitats i serveis relacionats amb el registre d'un usuari. **(Temps estimat: 10 hores)**

**T1.5.** Desenvolupament de la pantalla *Home*. Creació dels *layouts* i funcionalitats de la pantalla *Home*. **(Temps estimat: 10 hores)**

**T1.6.** Desenvolupament de la pantalla Notícies. Creació dels *layouts* i integrar les funcionalitats i serveis relacionats amb la pantalla de notícies. **(Temps estimat: 10 hores)**

**T1.7.** Desenvolupament de la pantalla Calendari. Creació dels *layouts* i integrar les funcionalitats i serveis relacionats amb el Calendari. **(Temps estimat: 10 hores)**

**T1.8.** Desenvolupament de la pantalla Incidències. Creació dels *layouts* i integrar les funcionalitats i serveis relacionats amb la pantalla d'incidències. **(Temps estimat: 10 hores)**

**T1.9.** Desenvolupament de la pantalla Enquestes. Creació dels *layouts* i integrar les funcionalitats i serveis relacionats amb la pantalla de Enquestes. **(Temps estimat: 10 hores)**

**T1.10.** Desenvolupament de la pantalla Configuració. Creació dels *layouts* i integrar les funcionalitats i serveis relacionats amb la pantalla de configuració. **(Temps estimat: 10 hores)**

**T1.11.** Testejar i buscar *bugs* relacionats amb les tasques anteriors i arreglar-los. Una de les tasques més importants consisteix en comprovar que tots els dissenys, funcionalitats i serveis són correctes i no tenen errors. **(Temps estimat: 10 hores)**

**T1.12.** Publicació de l'app a Google Play. **(Temps estimat: 10 hores)**

#### 5.1.2.- Desenvolupament de la base de dades

Per dur a terme aquestes tasques, necessitem recursos físics per poder emmagatzemar les bases de dades. En principi, es farà servir l'ordinador personal.

**T2.** Creació de la base de dades. **(Temps estimat: 15 hores)**

#### 5.1.3.- Desenvolupament de la pàgina web

En aquest cas, igual que en el desenvolupament Android, únicament és necessari un ordinador per poder desenvolupar el codi i poder executar-lo.

**T3.1.** Dissenyar el flux i dissenys de la pàgina web. **(Temps estimat: 10 hores)**

**T3.2.** Desenvolupament de la pàgina d'afegir esdeveniments. **(Temps estimat: 20 hores)**

**T3.3.** Desenvolupament de la pàgina de gestió d'incidències. **(Temps estimat: 20 hores)**

**T3.4.** Desenvolupament de la pàgina de gestió d'enquestes. **(Temps estimat: 20 hores)**

#### 5.1.4.- Desenvolupament dels serveis REST

Per les tasques següents, a part de l'ordinador per poder executar el codi, també és important tenir un programa anomenat *Postman* per simular crides REST al servidor i, d'aquesta forma, comprovar que els servies que estem programant funcionen correctament i tornen les dades esperades.

**T4.1.** Desenvolupament del servei *Login*. **(Temps estimat: 15 hores)**

**T4.2.** Desenvolupament del servei *Registre*. **(Temps estimat: 15 hores)**

**T4.3.** Desenvolupament del servei per descarregar notícies. **(Temps estimat: 15 hores)**

**T4.4.** Desenvolupament del servei de descarregar esdeveniments del calendari. **(Temps estimat: 15 hores)**

**T4.5.** Desenvolupament del servei de descarregar les incidències. **(Temps estimat: 15 hores)**

**T4.6.** Desenvolupament del servei d'enviar una incidència. **(Temps estimat: 15 hores)**

**T4.7.** Desenvolupament del servei de descarregar les enquestes. **(Temps estimat: 15 hores)**

**T4.8.** Desenvolupament del servei de respondre enquestes. **(Temps estimat: 15 hores)**

#### 5.1.5.- Gestió del projecte

Finalment, per la gestió del projecte únicament necessitem l'ordinador personal per poder anar documentant tot el procés. També és molt important la participació de la directora del projecte, Silvia Llorente Viejo, per guiar i donar feedback.

**T5.1.** Documentació. Redacció i descripció de tota la documentació relacionada amb el projecte: contextualització, planificació, pressupost i desenvolupament del projecte.  
**(Temps estimat: 75 hores)**

**T5.2.** Reunions de seguiment. **(Temps estimat: 10 hores)**

**T5.3.** Treball previ: entrevistes **(Temps estimat: 4 hores)**

**T5.4.** Gestió de projecte **(Temps estimat: 30 hores)**

**T5.5.** Anàlisi de tecnologies i de la competència **(Temps estimat: 10 hores)**

#### 5.2.- Seqüència lògica

És important planificar de forma lògica les tasques per tal d'optimitzar el temps al màxim. És important tenir creada la base de dades i els serveis per obtenir-ne les dades abans de començar a desenvolupar les tasques de les pantalles de l'aplicació Android. Per tant, tot i que comencem dissenyant el flux de l'aplicació, és important desenvolupar com abans millor els serveis i la base de dades. En segon lloc, es desenvolupa el pilar fonamental d'aquest projecte: l'aplicació Android. En aquests passos podem dur a terme tasques concurrentment ja que podem desenvolupar diferents funcionalitats alhora sense cap inconvenient. Finalment, desenvolupem la pàgina web que, en aquest cas, es desenvolupa de forma seqüencial ja que al ser un tema on no sóc expert, necessitaré més temps per acabar-ho.

### 5.3.- Taula resum

Codi	Tasca	Temps estimat	Dependències	Recursos (humans i materials)
T1.1.	Disseny App	8 hores		Project Manager, Dissenyador, Analista, despatx, ordinador
T1.2.	Projecte base	10 hores		Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.3.	<i>Login</i>	10 hores	T1.1., T1.2., T2, T4.1.	Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.4.	Registre	10 hores	T1.1., T1.2., T2, T4.2.	Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.5.	<i>Home</i>	10 hores	T1.1., T1.2., T2, T4.3., T4.4., T4.7.	Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.6.	Notícies	10 hores	T1.1., T1.2., T2, T4.3.	Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.7.	Calendari	10 hores	T1.1., T1.2., T2, T4.4.	Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.8.	Incidències	10 hores	T1.1., T1.2., T2, T4.5., T4.6.	Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.9.	Enquestes	10 hores	T1.1., T1.2., T2, T4.7., T4.8.	Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.10.	Configuració	10 hores	T1.1., T1.2.	Project Manager, Programador Android, despatx, ordinador, Android Studio i mòbil
T1.11.	Testejar app	10 hores	T1	Project Manager, Programador Android, despatx i mòbil
T1.12.	Publicació app	10 hores	T1	Project Manager, Programador Android, Despatx i ordinador
T2.	Bases de dades	15 hores		Project Manager, Programador back-end, ordinador i DB Browser SQL
T3.1.	Disseny web	10 hores		Project Manager, Dissenyador, despatx i ordinador.
T3.2.	Esdeveniments web	20 hores	T3.1.	Project Manager, Programador Web, despatx, ordinador, NetBeans
T3.3.	Incidències web	20 hores	T3.1.	Project Manager, Programador Web, despatx, ordinador, NetBeans

T3.4.	Enquestes web	20 hores	T3.1.	Project Manager, Programador Web, despatx, ordinador, NetBeans
T4.1.	Servei Login	15 hores	T2.	Project Manager, Programador back-end, despatx, ordinador, NetBeans i Postman
T4.2.	Servei Registre	15 hores	T2.	Project Manager, Programador back-end, despatx, ordinador, NetBeans i Postman
T4.3.	Descarregar Notícies	15 hores	T2.	Project Manager, Programador back-end, despatx, ordinador, NetBeans i Postman
T4.4.	Descarregar esdeveniments	15 hores	T2.	Project Manager, Programador back-end, despatx, ordinador, NetBeans i Postman
T4.5.	Descarregar incidències	15 hores	T2.	Project Manager, Programador back-end, despatx, ordinador, NetBeans i Postman
T4.6.	Enviar Incidència	15 hores	T2.	Project Manager, Programador back-end, despatx, ordinador, NetBeans i Postman
T4.7.	Descarregar enquestes	15 hores	T2.	Project Manager, Programador back-end, despatx, ordinador, NetBeans i Postman
T4.8.	Enviar resposta enquesta	15 hores	T2.	Project Manager, Programador back-end, despatx, ordinador, NetBeans i Postman
T5.1.	Documentació	75 hores		Becari, despatx, ordinador, word i tots els llibres descrits a bibliografia
T5.2.	Reunions	10 hores		Directora de projecte
T5.3.	Entrevistes	4 hores		Candidats a l'alcaldia de Badalona
T5.4.	Gestió del projecte	30 hores		Project Manager
T5.5.	Anàlisi del projecte	10 hores		Analista

Taula 1. Taula de tasques definides per hores, dependències i recursos.

## 5.4.- Estimacions temporals i Gantt

A la *figura 36* tenim el diagrama de Gantt corresponent a les tasques descrites a la taula anterior amb les dependències corresponents.

Hi ha tasques que són independents entre elles que podem executar de forma concurrent. En primer lloc, mentre pensem el disseny i el flux de l'aplicació també podem començar a preparar l'arquitectura del projecte Android.

Els serveis es poden fer de forma concurrent ja que no tenen dependència entre ells. El mateix passa amb les pantalles Android i per tant, també es poden executar de forma concurrent entre elles. A més a més, un cop acabada l'aplicació, s'ha de testejar i arreglar *bugs*. Finalment, el desenvolupament web al final que és la part més independent de tot el projecte.

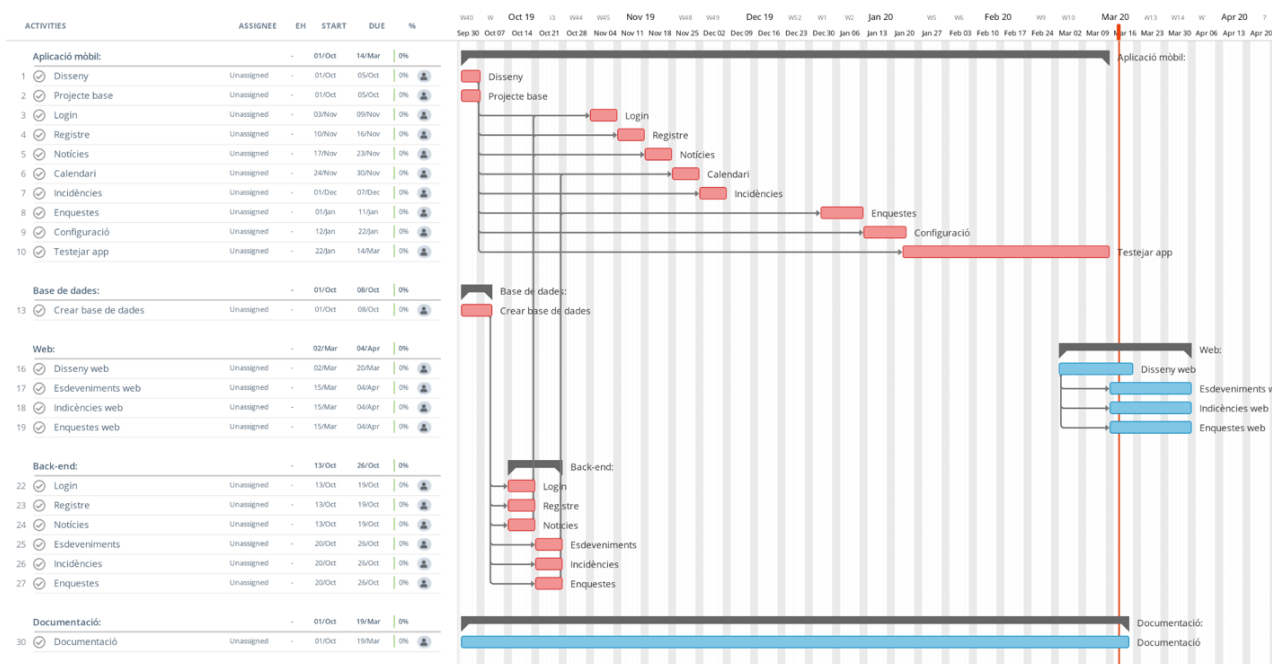


Figura 36. Diagrama de Gantt



### 5.5.- Desviacions i plans alternatius

Durant tot el procés de desenvolupament del projecte podem patir diferents problemes que ens portin a desviar-nos de la planificació establerta. És per això que la planificació està conclosa per tal d'acabar amb suficient temps per poder suportar aquestes desviacions.

En primer lloc, una possible desviació que trobem és que una tasca no compleixi el temps esperat i s'allargui o que apareguin subtasques necessàries pel desenvolupament correcte de l'aplicació que no s'havien contemplat prèviament.

En segon lloc, a l'estar poc familiaritzat amb el desenvolupament del servidor REST o la pàgina web, podria créixer la complexitat de les tasques i per tant, produir una desviació temporal de les tasques.

En tercer lloc, també és possible que ens encallem en el desenvolupament d'alguna tasca per la dificultat de trobar informació al respecte.

En quart lloc, que apareguin *bugs* a solucionar i no doni temps a arreglar-los a temps.

Finalment, problemes personals que puguin aparèixer i compliquin el temps de dedicació al Treball de Final de Grau.

Per evitar aquestes problemàtiques detectades, hi ha diversos plans alternatius:

- **Familiarització amb les tecnologies escollides:** un punt important és familiaritzar-nos abans de començar el desenvolupament amb les tecnologies escollides per tal d'evitar encallar-nos en un futur i haver de refer feina.
- **Oferir un marge de temps:** també és important fer una bona planificació per tal que ens quedin dies abans de l'entrega final per tal de tenir uns dies d'avantatge per tal de solucionar les desviacions patides.
- **Testejar durant desenvolupament:** per tal d'evitar possibles *bugs*, fer una revisió de les funcionalitats desenvolupades a mesura que anem desenvolupant les tasques per tal de facilitar la feina futura de testejar l'aplicació.
- **Code review:** un terme important en el desenvolupament de qualsevol software informàtic és fer una revisió de codi cada cop que es desenvolupa alguna tasca

per tal de detectar errors de codi abans de penjar els canvis. D'aquesta forma podem evitar possibles *bugs* futurs.

- **Sobreestimar les tasques:** afegir un temps de marge a les estimacions de les tasques per tal que, en cas que ens allarguéssim, no ens desviéssim pronunciadament de l'estimació.
- **Unit Tests:** tests inclosos al codi d'*Android* que s'utilitzen per tal de detectar errors de funcionalitat en el codi desenvolupat que a posteriori es convertiren en *bugs*.

## 5.6.- Desviacions patides durant el projecte

Tal i com esperava, s'han patit desviacions de temps a causa de pocs coneixements en l'àmbit de desenvolupament de la pàgina web i del servidor *REST Api*. A més a més, el desconeixement total en la fase de publicar el servidor i la pàgina web i arrancar-ho ha produït una desviació força gran.

A més a més, per motius personals i laborals no he pogut tenir una dedicació completa i això ha causat que la presentació d'aquest Treball de Final de Grau sigui durant el torn d'abril i no al de gener, tal i com s'esperava en un inici.

Finalment, una tasca que no es va contemplar en un inici a causa de que he adquirit coneixements d'aquest tema al llarg del projecte és en el disseny basat en una bona interfície i experiència d'usuari, de forma que ha causat que els dissenys de l'aplicació s'allarguessin més del que esperava.

## 6.- Gestió econòmica

### 6.1.- Pressupost

Aquest apartat mostra el pressupost que suposa aquest projecte. En primer lloc, s'identifiquen els costos necessaris per dur a terme el projecte. En segon lloc, es fa una estimació dels costos identificats, diferenciant costos directes de costos indirectes. Es tindran en compte les tasques i les estimacions temporals vistes a l'apartat anterior per tal d'estimar els costos de les tasques.

#### 6.1.1.- Identificació dels costos

Per dur a terme un pressupost, primer s'han d'identificar tots els costos que podem trobar-nos al llarg del projecte per tal de, posteriorment, fer unes estimacions basant-nos amb les tasques, material o lloc de treball que necessitem.

En primer lloc, les estimacions dels treballadors necessaris per dur a terme tota la part tecnològica. Suposarem que tots els treballadors siguin de la categoria *Junior*, és a dir, tinguin entre 0 i 2 anys d'experiència [32]:

- **Dissenyador gràfic:** persona encarregada a dissenyar segons els patrons d'interfície d'usuari adients per executar l'aplicació. El preu mitjà d'un dissenyador d'aplicacions mòbils a Barcelona és de 17 € per hora.
- **Programador Android:** programador que s'encarrega de desenvolupar l'aplicació mòbil pel sistema operatiu Android. El preu mitjà a Barcelona és de 19 € per hora.
- **Programador web:** programador web per desenvolupar la pàgina web d'administradors. El preu mitjà a Espanya per un desenvolupador web és de 15 € per hora.
- **Programador back-end:** desenvolupador encarregat de programar els diferents serveis necessaris per connectar la base de dades amb l'aplicació i la web. El preu mitjà a Barcelona és de 15 € per hora.

- **Becari:** també tindrem en compte el perfil d'un becari que pot ajudar a documentar el projecte. Un estudiant de pràctiques de la Universitat Politècnica de Catalunya cobra per conveni universitari 8 € per hora.
- **Project manager:** encarregat de la gestió i organització del projecte. El preu aproximat d'un *project manager* és de 25€ per hora.
- **Analista:** encarregat d'investigar la competència i les tecnologies més òptimes per desenvolupar el projecte. El preu és de 20€ per hora.

En segon lloc, després de definir els diferents actors necessaris per desenvolupar el projecte, també hem d'identificar els costos genèrics d'espai, electricitat i material, entre d'altres:

- **Espai:** el preu d'un coworking a Barcelona s'aproxima als 160 € al mes per treballador. Suposem que es pot fer un lloguer durant un quadrimestre per dues persones, de forma que, seguint el diagrama de Gantt, podrà treballar un dels programadors i el becari alhora. Seguint la planificació, els programadors s'alternaran aquest espai de treball.
- **Electricitat:** amb el lloguer d'un despatx seria necessari però, amb el lloguer d'un coworking, aquest cost no cal tenir-lo en compte ja que acostuma a estar inclòs en el preu.
- **Equips:** necessitem un equip potent, per tant, suposem que comprem dos ordinadors per tal de poder combinar-los segons la planificació temporal. Ens decantem per un Macbook pro de 128GB, amb un preu de venda al públic de 1.499 €. S'ha de tenir en compte el preu d'amortització, que es calcula amb la següent fórmula:  $\text{Amortització} = \text{Preu de compra} / (\text{anys d'amortització} * \text{hores laborables anuals}) * \text{hores d'ús durant TFG}$ . Suposem que els amortitzem 4 anys i que l'any té 2.000 hores laborables anuals de les quals 412 seran dedicades al TFG. Per tant, el preu dels equips és de:  $\text{Amortització} = 2.998 \text{ €} / (4 \text{ anys d'amortització} * 2.000 \text{ hores anuals laborables}) * 412 \text{ hores d'ús pel TFG} = 154,39 \text{ €}$

- **Mòbil de prova:** per poder testejar correctament l'aplicació necessitem un mòbil de prova. Suposem que comprem un mòbil de gama mitja com el Samsung A50 de 299 €. Pel que fa al mòbil, s'aplica la mateixa fórmula que amb els equips suposant que té 2 anys d'amortització, que les hores laborables anuals són 2.000 i que s'utilitzarà el dispositiu mòbil 110 hores pel TFG. Llavors, el preu del mòbil de prova és de:  $\text{Amortització} = 299\text{€} / (2 \text{ anys d'amortització} * 2.000 \text{ hores anuals laborables}) * 110 \text{ hores d'ús pel TFG} = 8,22 \text{ €}$
- **Software:** en un principi, tots els programes necessaris per desenvolupar l'aplicació, la web i el servidor són gratuïts. Únicament necessitem comprar l'Adobe Photoshop per poder dissenyar les pantalles. El preu és de 24,19 € al mes.

#### 6.1.2.- Estimació dels costos

A continuació es mostren diferents taules segons el tipus de cost, sigui directe o indirecte. En primer lloc, trobarem les activitats descrites al diagrama de Gantt amb el pressupost estimat, basant-nos amb els preus descrits al punt anterior.

Pel càlcul del pressupost, s'han de tenir presents diferents factors. Tal i com es comenta al punt anterior, s'utilitzen els preus mitjans per hora de l'actor que durà a terme la tasca. Tot i això, és important tenir en compte el cost de la seguretat social que, aproximadament, és el sou brut multiplicat per 1,35. Aquest cost s'anomena *Cost per Activitat (CPA)*. També és important tenir en compte el preu dels recursos necessaris per treballar, on s'ha de tenir present l'espai per treballar i el material. Això s'anomena *Cost Genèric (CG)*. En tercer lloc, també s'ha de valorar el cost de contingència, que és el cost genèric d'obstacles no previstos i s'acostuma a calcular com un 15% del *Cost per Activitat* i el *Cost Genèric*. Finalment, cal tenir en compte els possibles riscos, on s'ha de valorar el percentatge de risc pel càlcul del seu cost. Aquests riscos es valoren segons el percentatge de risc i el cost de l'activitat.

Per fer-ho més fàcil visualment, les tasques executades per diferents actors i els costos, diferenciant directes o indirectes, estan diferenciats per colors.

#### *6.1.2.1- Costos directes*

Els costos directes són els que estan directament relacionats amb el producte o la seva execució. En el nostre cas, aquests costos consisteixen en la mà d'obra, és a dir, els programadors i dissenyador. A continuació, veurem la taula amb els costos directes:

Activitat	Hores totals	Preu per hora (€)	Import total (€)	Observacions
Disseny App	8 hores	17	136	Preu brut sense seguretat social
Projecte base	10 hores	19	190	Preu brut sense seguretat social
<i>Login</i>	10 hores	19	190	Preu brut sense seguretat social
Registre	10 hores	19	190	Preu brut sense seguretat social
<i>Home</i>	10 hores	19	190	Preu brut sense seguretat social
Notícies	10 hores	19	190	Preu brut sense seguretat social
Calendari	10 hores	19	190	Preu brut sense seguretat social
Incidències	10 hores	19	190	Preu brut sense seguretat social
Enquestes	10 hores	19	190	Preu brut sense seguretat social
Configuració	10 hores	19	190	Preu brut sense seguretat social
Testejar app	10 hores	19	190	Preu brut sense seguretat social
Publicació app	10 hores	19	190	Preu brut sense seguretat social
Disseny web	10 hores	15	150	Preu brut sense seguretat social

Esdeveniments web	20 hores	15	300	Preu brut sense seguretat social
Incidències web	20 hores	15	300	Preu brut sense seguretat social
Enquestes web	20 hores	15	300	Preu brut sense seguretat social
Bases de dades	15 hores	15	225	Preu brut sense seguretat social
Servei Login	15 hores	15	225	Preu brut sense seguretat social
Servei Registre	15 hores	15	225	Preu brut sense seguretat social
Descarregar Notícies	15 hores	15	225	Preu brut sense seguretat social
Descarregar esdeveniments	15 hores	15	225	Preu brut sense seguretat social
Descarregar incidències	15 hores	15	225	Preu brut sense seguretat social
Enviar Incidència	15 hores	15	225	Preu brut sense seguretat social
Descarregar enquestes	15 hores	15	225	Preu brut sense seguretat social
Enviar resposta enquesta	15 hores	15	225	Preu brut sense seguretat social
Documentació	75 hores	8	600	Preu brut sense seguretat social
Gestió del projecte	30 hores	25	750	Preu brut sense seguretat social



Anàlisi del projecte	10 hores	20	200	Preu brut sense seguretat social
I1: Bugs de l'aplicació mòbil (risc 60%)			1.335,6	Risc multiplicat pel cost.
I2: Bugs back-end (risc 60%)			1.215	Risc multiplicat pel cost.
I3: Bugs d'aplicació web (risc 30%)			315	Risc multiplicat pel cost.

*Taula 2. Taula de costos en relació amb les tasques.*

#### 6.1.2.2.- Costos indirectes

A diferencia dels costos directes, els costos indirectes són aquells costos que no es poden assignar directament al desenvolupament del producte final o, en cas d'una empresa, que té un caràcter més general dins d'aquesta. En aquest cas, els costos indirectes són el lloguer de l'espai, electricitat, cost dels equips i del material. A continuació, veurem la taula amb els diferents costos indirectes desglossats:

Activitat	Hores totals	Preu per hora (€)	Import total (€)	Observacions
Cost d'espai	4 mesos	160 €/mes per persona	1.280	Lloguem durant 4 mesos per dues persones
Cost electricitat	4 mesos	0	0	Inclòs en el preu del coworking
Cost equips		1.499 €/u	154,39	Dues unitats per poder treballar concurrentment
Cost mòbil		299 €/u	8,22	Necessari per testejar l'aplicació
Cost Software		0	0	El software necessari és gratuït

Taula 3. Taula de costos indirectes.

#### 6.1.3.- Cost total

Un cop vistes les estimacions de les tasques segons els sous i preus del mercat actual a Barcelona, procedim al càlcul total del pressupost. En primer lloc veurem el cost total per seccions. En segon lloc, calculem el Cost Per Activitat, que és calcula multiplicant el sou brut per 1,35. En tercer lloc, el Cost Genèric, que és la suma total dels costos indirectes. Finalment, cal tenir en compte els imprevistos, obstacles no previstos i arreglar errors que trobem quan passem el procés de proves amb el servidor back-end,

aplicació Android i web. Es calcula primer la contingència, que és el 15% del Cost per Activitat i el Cost Genèric, tal i com es comenta a l'inici de la secció i, després, el cost total de les incidències.

Activitat	Preu total
Aplicació Android	2.226,00 €
Pàgina web	1.050,00 €
Back-end	2.025,00 €
Gestió de projecte	1.550,00€
Costos indirectes	1.442,61 €
Cost Per Activitat	9.248,85 €
Cost Genèric	1.442,61 €
Contingència	1.603,71 €
Incidències	2.865,60 €
<b>Cost Total</b>	<b>15.160,77 €</b>

*Taula 4. Taula de costos totals.*

## 7.- Identificació de lleis i regulacions

En aquest projecte s'han contemplat dues regulacions que s'han de tenir en compte a l'hora de desenvolupar el projecte dins del marc legal. Aquestes dues regulacions estan relacionades amb la protecció de dades i amb drets d'autor.

### 7.1.- Llei orgànica de protecció de dades i garantia de drets digitals.

Al maig de 2016 va ésser aprovada la regulació coneguda com a GDPR, que és l'acrònim en anglès de *“Regulació de protecció de dades”*, per la Comissió Europea. Els principals canvis d'aquesta regulació són que qualsevol empresa que recopili dades personals, necessitarà el consentiment explícit i informat de la persona. Aquesta persona també tindrà el dret de demanar en qualsevol moment totes les seves dades personals. A més a més les multes per l'incompliment d'aquesta llei s'agreguen fins a un 4% del a facturació global de l'empresa castigada. Finalment, s'han ampliat els requisits per aconseguir dades de segona mà. Abans d'aquesta regulació, la majoria d'empreses podien vendre les dades dels seus usuaris a tercers, de forma que tots dos es beneficiaven de les dades personals dels usuaris. Des de l'aplicació d'aquesta normativa hi ha una sèrie de requisits que s'han de complir per tal de poder cedir les dades a tercers. [33]

El desembre de 2018, a Espanya va aparèixer la *“Llei Orgànica de Protecció de Dades i Garantia de Drets Digitals”* com una adaptació de la normativa europea. Aquesta regulació obliga a ampliar la informació que s'ha de donar als usuaris sobre el tractament de les seves dades personals. També, tal i com obliga la llei europea, és obligat que es preguntí explícitament que un usuari cedeix les dades. Pel que fa al consentiment de menors, per obtenir les dades de qualsevol menor de 14 anys, es necessita el permís dels pares o tutor legal. Finalment, hi ha una nova clàusula que permet a qualsevol usuari que pugui demanar que s'esborrin totes les seves dades de qualsevol empresa quan revoqui el consentiment. [34]

## 7.2.- Llei de propietat intel·lectual

Aquestes regulacions és necessari tenir-les en compte ja que les notícies mostrades a la secció pertinent, seran extretes de diferents diaris o revistes tal de poder mantenir als usuaris informats.

Segons l'article 33 de la llei de propietat intel·lectual, un article o treball sobre temes d'actualitat difosos per medis de comunicació social poden ser difosos i reproduïts públicament, sempre i quan es citi la font i l'autor del mateix. [35]

## 8. Informe de sostenibilitat

### 8.1. Autoavaluació

Personalment crec que, en general, els enginyers acostumen a estar poc conscienciats de la importància de la sostenibilitat. També crec que al llarg del grau d'enginyeria, tot i remarcar a diferents assignatures la importància de la sostenibilitat, es donen poques eines i poca càrrega lectiva a aquesta part. Per tant, la majoria d'estudiants no hi posen gaire esforç.

Una bona forma de conscienciar de la importància de la sostenibilitat seria avaluar més treballs sobre sostenibilitat amb més pes a la nota final de l'assignatura, ja que correspondre únicament a la competència transversal no és donar la importància necessària al tema.

### 8.2. Dimensió econòmica

El cost total estimat és de 15.160,77 €, un preu totalment raonable per un projecte d'aquestes dimensions. Donat que aquest projecte facilitarà la vida del ciutadà i la de l'Ajuntament, no és un preu massa elevat.

Actualment hi ha poques solucions semblants al mercat pel nostre públic i, tot i que si que és cert que hi ha ciutats mitjanes amb aplicacions, el seu manteniment és igual de car i no dóna tantes opcions ni facilitats com aquest projecte.

Amb aquesta aplicació possiblement l'Ajuntament no reduirà costos, però tindrà molt més accessible l'opinió del ciutadà i tindrà més oportunitats d'actuar a l'hora de mantenir la ciutat, de forma que això dóna un cost elevat a aquest projecte.

### 8.3. Dimensió ambiental

Aquest projecte també ofereix l'opció d'arreglar i mantenir la ciutat i també, de que els ciutadans opinin. D'aquesta forma, es poden oferir propostes favorables al medi ambient per millorar la ciutat en funció del que els ciutadans demanen.

També ajuda en que, quan hi hagi esdeveniments, ajudarà a l'Ajuntament a reduir el consum de materials per anunciar esdeveniments ja que la forma més ràpida i fàcil serà via l'aplicació.

Sí que és cert que en el propi projecte es podria intentar reduir el consum utilitzant productes de segona mà com per exemple els ordinadors o el telèfon mòbil de prova. Tot i això, una de les opcions més factibles és comprar-ne de nous per tenir la certesa d'una alta qualitat per no tenir pèrdues temporals a l'hora d'executar el codi. A posteriori, podrem reutilitzar aquestes màquines per no contaminar.

Actualment al mercat no hi ha res semblant que faciliti de tal forma la nostra proposta, de manera que és una bona forma de començar.

### 8.4. Dimensió social

A nivell personal aquest projecte m'ajudarà a créixer professionalment ja que he de dur a terme tot un projecte executant tots els rols. A part, m'ajudarà a aprendre i millorar les parts de desenvolupament que més costoses són per mi, com per exemple la programació web i *back-end*. A part, també m'ajudarà a millorar i aprendre més coneixements sobre el meu camp, la programació Android.

Pel que fa el projecte, crec que és una molt bona solució per facilitar la vida dels ciutadans, ja que apropar la comunicació amb l'Ajuntament per millorar tots els problemes que la gent troba al seu dia a dia. També ajuda a tota la població a estar informada del que passa a la seva ciutat.

La necessitat del projecte és real, ja que amb l'avanç tecnològic que hi ha actualment és essencial que qualsevol municipi estigui dotat d'una aplicació que faciliti la vida a la gent.

A part, de mica en mica es poden incorporar més funcionalitats si hi ha un manteniment constant.



## 9.- Conclusions

### 9.1.- Dificultats i problemes

Durant l'execució d'aquest projecte han aparegut diferents dificultats i problemàtiques que han endarrerit l'execució del mateix. A continuació, es descriu una llista de les problemàtiques que han aparegut al llarg del projecte i quin ha sigut el motiu:

- Una problemàtica que ja estava contemplada és que a l'estar especialitzat en una àrea com el *front-end*, on no s'ha de desplegar cap servidor, ha fet costós el desenvolupament del servidor *back-end* i el seu desplegament.
- Tot i que a la Universitat ens ensenyen les bases per qualsevol llenguatge i la facilitat en aprendre'n que adquirim al llarg del grau, sempre implica un gran cost temporal aprendre un llenguatge nou.
- Donat que durant el Grau només es treballa amb bases de dades relacionals, la posada en pràctica de les bases de dades no relacionals ha causat un endarreriment en l'execució de l'entrega.
- Degut a actualitzacions de versions de llibreries, va sorgir una incompatibilitat entre elles i es va haver de buscar les versions que eren compatibles. Resoldre compatibilitats de llibreries sempre és una tasca complicada.
- Després d'intentar posar-me en contacte amb els candidats a l'alcaldia de Badalona he vist que és més complicat del que pensava posar-se en contacte amb líders polítics.
- Degut a problemes personals d'un canvi laboral, la dedicació al projecte no va poder ser l'adequada. Com a conseqüència el projecte s'ha endarrerit.
- Cap portal de notícies de Badalona tenia les notícies disponibles per poder afegir a l'aplicació mòbil i s'ha optat per una alternativa.

## 9.2. Conclusions dels objectius inicials

Tot i les dificultats i problemes que han anat sorgint, els quatre objectius s'han pogut elaborar amb èxit: aplicació Android, servidor back-end, aplicació web i base de dades.

Pel que fa a l'aplicació Android, s'ha pogut desenvolupar amb èxit amb totes les seves funcionalitats descrites. Tot i això, la secció de notícies no s'ha pogut desenvolupar tal i com es va definir en un inici donat que cap portal de notícies ni diari de Badalona oferia les seves notícies per poder afegir a l'aplicació Android. A causa d'això, es va haver de buscar una alternativa donat que el portal de notícies es considera una secció rellevant en aquest projecte. L'única manera amb la que es va poder fer va ser agafant els *tuits* dels diferents diaris a *Twitter*. Donat que l'API de *Twitter* té les seves limitacions, es va haver d'afegir un apartat per tal que l'usuari escollís la seva font favorita per tal de mostrar els *tuits* d'aquesta font. Tot i que no era l'objectiu inicial, s'ha rectificat amb èxit.

## 9.3.- Conclusions personals

Un cop acabat aquest treball, puc dir que estic satisfet per diferents motius. En primer lloc, considero que aprendre a treballar de forma autònoma és un punt clau i molt important, tant per ser autosuficient com per aprendre a valorar la feina que moltes vegades no et pertoca en un projecte. En segon lloc, aprendre nous llenguatges és sempre satisfactori i amb el desenvolupament d'aquest projecte, he hagut de treballar a fons amb llenguatges que desconeixia totalment. En tercer lloc, l'aprenentatge de diferents tipus de bases de dades i de com gestionar-les és un punt important. Finalment, tot i que en la majoria de perfils d'informàtics no és una feina que acostumi a agradar, considero que aprendre a estructurar i documentar correctament un projecte és una tasca molt important.

En finalitzar aquest treball m'emporto una bona experiència i molts coneixements nous. A més, després de cometre diferents errors al llarg del projecte he après a com contrarestar-los per millorar dia a dia.

## 9.4.- Conclusions de competències tècniques

### 9.4.1.- CTI1.1

Demostrar comprensió de l'entorn d'una organització i de les seves necessitats en l'àmbit de les tecnologies de la informació i les comunicacions. **[En profunditat]**

En aquest projecte s'han analitzat les diferències entre les ciutats mitjanes i la resta de ciutats per tal de poder definir correctament les seves mancances i necessitats per tal de poder oferir una solució òptima per iniciar-se, dins del possible abast d'aquest projecte, en el denominat ciutat intel·ligent.

Després d'aquesta anàlisi s'han buscat les tecnologies més òptimes per plantejar-ho i executar-ho de forma que la comunicació entre serveis i aplicacions sigui efectiva.

### 9.4.2.- CTI1.3

Seleccionar, desplegar, integrar i gestionar sistemes d'informació que satisfacin les necessitats de l'organització amb els criteris de cost i qualitat identificats. **[Bastant]**

En aquest projecte s'han plantejat diferents tecnologies i s'han seleccionat les més adequades, s'han executat i s'han integrat en el projecte seguint unes pautes de cost establertes a l'apartat 6.

### 9.4.3.- CTI3.1

Concebre sistemes, aplicacions i serveis basats en tecnologies de xarxa, tenint en compte Internet, web, comerç electrònic, multimèdia, serveis interactius i computació ubiqua. **[Bastant]**

Les bases d'aquest treball són el desenvolupament d'una aplicació mòbil, una aplicació web, un servidor i una base de dades connectats a través d'internet i tecnologies de xarxa.

#### 9.4.4.- CTI4

Emprar metodologies centrades en l'usuari i l'organització per al desenvolupament, l'avaluació i la gestió d'aplicacions i sistemes basats en tecnologies de la informació que assegurin l'accessibilitat, l'ergonomia i la usabilitat dels sistemes. **[En profunditat]**

Pel desenvolupament s'han utilitzat metodologies àgils per tal de poder executar el projecte de la forma més efectiva i amb més reacció. Després de cada *sprint* es genera una entrega per tal de poder provar les funcionalitats integrades i el correcte funcionament. També es comprova que sigui usable i útil per l'usuari i, si no fos així, al següent *sprint* s'afegeixen tasques noves per millorar o refer el que no sigui correcte.

## 10.- Vocabulari

- **Activity:** interfície que interactua amb l'usuari on es pot posar una finestra interactiva personalitzada. Normalment ocupa tota la pantalla del dispositiu.
- **Android:** sistema operatiu desenvolupat per Google el 2007 basat en Linux.
- **back-end:** capa encarregada de gestionar les dades. Rep informació i en retorna una resposta a partir d'aquesta.
- **Boolean:** tipus de dades binari. Únicament pot tenir el valor true o false.
- **Bug:** error en un software informàtic.
- **Controller:** capa encarregada de comunicar les altes capes existents.
- **Coworking:** espai on diferents empreses o autònoms poden llogar un espai per treballar.
- **Date:** tipus de dada que representa una data i hora.
- **Framework:** conjunt de conceptes i pràctiques que serveix com a referència per encarar i resoldre problemes.
- **front-end:** capa de presentació que interactua amb l'usuari.
- **Headers:** paràmetres bàsics d'una crida a un servei, com podria ser format de dades o clau d'accés
- **iOS:** sistema operatiu desenvolupat per Apple Inc. l'any 2007.
- **Layout:** fitxer que defineix una estructura d'interfície d'usuari amb diferents elements.
- **Model:** capa d'una aplicació que representa unes dades concretes d'un element.
- **Number:** tipus de dada que representa els nombres enters en llenguatge TypeScript
- **Open source:** model de desenvolupament de software basat en una col·laboració oberta.

- **Presenter:** capa encarregada de comunicar les altres capes existents. Mateixa funcionalitat que el Controller amb la diferència que hi ha un Presenter per a cada Activity, de forma que s'evita tenir classes molt llargues.
- **Refactor:** tècnica d'enginyeria del software que consisteix en reestructurar un codi font.
- **Request:** dades que s'envien a un servei.
- **Response:** dades que retorna un servei.
- **REST Api:** interfície que utilitza el protocol HTTP per obtenir o generar dades.
- **Smartphone:** tipus d'ordinador de butxaca amb capacitats de telèfon mòbil.
- **Sprint:** interval de temps prefixat durant el qual es crea un increment del codi utilitzable i que es pugui entregar.
- **SQLite:** sistema de gestió de bases de dades relacionals.
- **String:** tipus de dades que representa una cadena de caràcters.
- **Test unitari:** forma de comprovar el correcte funcionament d'una unitat de codi.
- **Thread:** seqüència de tasques encadenades que poden ser executades per un sistema operatiu.
- **Token:** clau d'accés per accedir a un servidor REST Api concret.
- **Toolbar:** barra d'eines. En general, barra amb diverses funcionalitats que hi ha a la part superior de les aplicacions mòbil.
- **Twitter:** xarxa social creada el 2006 on els usuaris poden penjar escrits i interactuar amb ells.
- **View:** capa encarregada de gestionar les interaccions de l'usuari.

## 11.- Referències

- [1] Muñoz, R. (2019, 26 febrer). Dos tercios de la humanidad ya usan un móvil. *El País*.
- [2] Rojas, A. (2009, Octubre). Ciudades medias y expansión territorial. *La Ciudad viva*.
- [3] Llop, J.; Iglesias, B.; Vargas R.; Blanc F. (2019, Abril). Las ciudades intermedias: concepto y dimensiones. *CIUDADES* n22. Universidad de Valladolid. ISSN-E: 2445-3943.
- [4] Cuidem Terrassa. URL:  
<https://play.google.com/store/apps/details?id=com.idasasistemas.terrassacuidem&gl=ES>
- [5] Girona App. URL:  
<https://play.google.com/store/apps/details?id=com.appsproyectos.girona&gl=ES>
- [6] MacDonald, M. (2016, 20 Desembre). The Importance of Market Segmentation.
- [7] Facultat d'Informàtica de Barcelona, UPC. (2017, 18 Octubre) Normativa del treball de final de grau del grau en Enginyeria Informàtica de la Facultat d'Informàtica de Barcelona.
- [8] JIRA. About us. URL: <https://www.atlassian.com/company>
- [9] Trello. Trello Tour. URL: <https://trello.com/tour>
- [10] Git. About Git. URL: <https://git-scm.com/about>
- [11] GitKraken. GitKraken Support. URL: <https://support.gitkraken.com/>
- [12] Sourcetree. Get started with SourceTree. URL:  
<https://confluence.atlassian.com/get-started-with-sourcetree>
- [13] Leiva, A. Clean Architecture for Android with Kotlin: a pragmatic approach for starters. URL: <https://antonioleiva.com/clean-architecture-android/>
- [14] Leiva, A (2019, 11 Febrer). Kotlin For Android Developers.
- [15] IntelliJ Idea. IntelliJ Idea support. URL: <https://intellij-support.jetbrains.com/hc/en-us#>

- [16] Google Android Developers. URL: <https://developer.android.com/>
- [17] JetBrains. Kotlin 2019 – The State of Developer Ecosystem in 2019 Infographic. URL: <https://www.jetbrains.com/lp/devecosystem-2019/kotlin/>
- [18] Harrison T. (2018, 30 Gener). Realm vs Room for Android. URL: <https://www.oakcity.io/blog/2018/01/30/realm-vs-room-android/>
- [19] TypeScript Overview. URL: [https://www.tutorialspoint.com/typescript/typescript\\_overview.htm](https://www.tutorialspoint.com/typescript/typescript_overview.htm)
- [20] Spring Boot. Spring Boot Support. URL: <https://spring.io/projects/spring-boot>
- [21] NetBeans. NetBeans Support. URL: <https://netbeans.org/features/>
- [22] VisualStudio. VisualStudio Support. URL: <https://visualstudio.microsoft.com/es/?rr=https%3A%2F%2Fdocs.microsoft.com%2Fen-us%2Fvisualstudio%2F%3Fview%3Dvs-2019>
- [23] Python. About. URL: <https://www.python.org/about/>
- [24] JavaScript. About. URL: <https://www.javascript.com/about>
- [25] PHP. About. URL: <https://www.php.net/manual/es/intro-what-is.php>
- [26] Viviana (2018, 29 Octubre). MySQL vs MongoDB. URL: <https://guiadev.com/mysql-vs-mongodb/>
- [27] MySQLWorkbench. MySQLWorkbench Documentation. URL: <https://dev.mysql.com/doc/workbench/en/>
- [28] dbWatch. dbWatch Database Monitoring and Management. URL: <https://www.dbwatch.com/learn-more>
- [29] MongoDB Compass. MongoDB Compass Tools. URL: <https://docs.mongodb.com/tools/>
- [30] Sastre, A. Dropbox vs Google Drive: ¿Cuándo es mejor usar cada uno en tu negocio online? URL: <https://marketinglibelula.com/dropbox-vs-google-drive/>
- [31] The Standish Group. (2012) *Chaos Manifesto*.



[32] Open Webinars (2018). Guía de salarios IT.

[33] Brandom, R. (2018, 28 Març). How Europe's new privacy rule is reshaping the internet. *The Verge*.

[34] Consultoría Integral Tecnológica. (2018) Ley de Protección de Datos y Garantía de Derechos Digitals. URL: <https://protecciondatos-lopd.com/empresas/nueva-ley-proteccion-datos-2018/>

[35] Llei de Propietat Intel·lectual, Article 33. Trabajos sobre temas de la actualidad.

